Unsupervised Models for Morpheme Segmentation and Morphology Learning

MATHIAS CREUTZ and KRISTA LAGUS
Helsinki University of Technology

We present a model family called Morfessor for the unsupervised induction of a simple morphology from raw text data. The model is formulated in a probabilistic maximum a posteriori framework. Morfessor can handle highly inflecting and compounding languages where words can consist of lengthy sequences of morphemes. A lexicon of word segments, called *morphs*, is induced from the data. The lexicon stores information about both the usage and form of the morphs. Several instances of the model are evaluated quantitatively in a morpheme segmentation task on different sized sets of Finnish as well as English data. Morfessor is shown to perform very well compared to a widely known benchmark algorithm, in particular on Finnish data.

Categories and Subject Descriptors: I.2.7 [**Artificial Intelligence**]: Natural Language Processing; I.2.6 [**Artificial Intelligence**]: Learning—*Language acquisition*; I.5.1 [**Pattern Recognition**]: Models—*Statistical*; H.1.1 [**Models and Principles**]: Systems and Information Theory—*Information theory*

General Terms: Algorithms, Experimentation, Languages, Performance

Additional Key Words and Phrases: Efficient storage, highly inflecting and compounding languages, language independent methods, maximum a posteriori (MAP) estimation, morpheme lexicon and segmentation, unsupervised learning

ACM Reference Format:

Creutz, M. and Lagus, K. 2007. Unsupervised models for morpheme segmentation and morphology learning. ACM Trans. Speech Lang. Process. 4, 1, Article 3 (January 2007), 34 pages. DOI = 10.1145/1187415.1187418 http://doi.acm.org/10.1145/1187415.1187418.

1. INTRODUCTION

When constructing a system that is capable of understanding and producing language, a fundamental task is the determination of the basic language units and their relationships. Many practical natural language processing (NLP)

We would like to thank the Graduate School of Language Technology in Finland for funding part of this work, and we express our gratitude to the Finnish IT Center for Science and the Finnish National News Agency for providing text corpora.
Authors' address: Helsinki University of Technology, Adaptive Informatics Research Centre, P.O. Box 5400, FIN-02015 HUT, Finland; email: mathias.creutz@hut.fi.

problems are best solved using lexical resources, in their simplest form an application-specific vocabulary. For example, in information retrieval, the analysis entails collecting a list of words and detecting their association with topics of discussion. Moreover, a vocabulary is essential for obtaining good results in speech recognition.

Words are often thought of as basic units of representation. However, especially in inflecting and compounding languages this view is hardly optimal. For instance, if one treats the English words hand, hands, and left-handed as separate entities, one neglects the close relationships between these words as well as the relationship of the plural s to other plural word forms e.g., heads, arms, fingers. Overlooking these regularities accentuates data sparsity which is a serious problem in statistical language modeling.

According to linguistic theory, morphemes are the smallest meaning-bearing units of language as well as the smallest units of syntax [Matthews 1991]. Every word consists of one or several morphemes; consider, for instance, the English words hand, hand+s, left+hand+ed, finger+s, un+avail+able. There exist linguistic methods and automatic tools for retrieving morphological analyses for words, for example, based on the two-level morphology formalism [Koskenniemi 1983]. However, these systems must be tailored separately for each language, which demands a large amount of manual work by experts. Moreover, specific tasks often require specialized vocabularies which must keep pace with the rapidly evolving terminologies.

If it is possible to discover a morphology automatically from unannotated text, language and task independence are easier to achieve. As we will demonstrate in this work, by observing the language data alone, it is possible to come up with a model that captures regularities within the set of observed word forms. If a human were to learn a language in an analogous way, this would correspond to being exposed to a stream of large amounts of language without observing or interacting with the world where this language is produced. This is clearly not a realistic assumption about language learning in humans. However, Saffran et al. [1996] show that adults are capable of discovering word units rapidly in a stream of a nonsense language without any connection to meaning. This suggests that humans do use distributional cues, such as transition probabilities between sounds, in language learning. And these kinds of statistical patterns in language data can be successfully exploited by appropriately designed algorithms.

Based on a comprehensive review of contemporary studies of how children start to acquire language, Kit [2003] concludes that children certainly make use of statistical cues. Kit further proposes the least-effort principle as a probable underlying approach that is supported by both empirical evidence and theoretical considerations. The least-effort principle corresponds to Occam's razor, which says that among equally performing models one should prefer the smallest one. This can be formulated mathematically using the Minimum Description Length (MDL) principle [Rissanen 1989] or in a probabilistic framework as a maximum a posteriori (MAP) model.

Generally, a system using language benefits from representing as large a vocabulary as possible. However, both humans and artificial systems need to be able to store language economically using limited memory capacity. This is particularly true for small portable devices. For example, if one has 500,000 word forms in a statistical n-gram language model, or essentially the same information using only 20,000 morphemes, considerable improvements in efficiency can be obtained.

In language understanding and generation, one must not only represent possible word forms but also their rules of generation in the context of other words. An important consideration is the ability to generate and to recognize unseen word forms and expressions. For example, we would expect a system to be able to handle the word shoewiping when some other related word forms have been observed (e.g., shoe, wiped). If a word-based system has not observed a word, it cannot recognize or generate it. In contrast, a morpheme-based system can generate and recognize a much larger number of different word forms than it has observed.

In this work, we describe a general probabilistic model family for morphology induction. The model family that we call *Morfessor* consists of independent components that can be combined in different configurations. We utilize the maximum a posteriori framework for expressing the model optimization criteria.

Morfessor segments the input words into units called *morphs*. A lexicon of morphs is constructed where information about both the distributional nature (usage) and form of each morph is stored. *Usage* relates to the distributional nature of the occurrence of morphs in words. *Form* corresponds to the string of letters that comprise the morph. We experimentally evaluate different instances of the Morfessor model and compare them against a benchmark morphology-learning algorithm [Goldsmith 2001, 2005].

1.1 Structure of the Article

Related work on both morphology learning and word segmentation is discussed in Section 2. Moreover, the point of view of applying different mathematical modeling frameworks is also considered.

The Morfessor model family is outlined in Section 3. The components of the model as well as their interpretations in terms of usage and form are discussed in detail. A summary of our previous morphology discovery methods as instances of this general framework is presented in Section 4.

Section 5 presents thorough experimental results, comparing the different instances of the model with datasets of different sizes, ranging from thousands to millions of words. The results are intended to provide an understanding on how particular components of the general model affect morphology learning. We use an evaluation task that measures segmentation accuracy and coverage of the proposed segmentations against gold standard segmentations for Finnish and English.

Section 6 discusses issues beyond the discovery of morpheme boundaries as well as considering aspects that are not handled by the current model framework. Conclusions are presented in Section 7.

2. RELATED WORK

Unsupervised morphology induction is closely connected with the field of automatic word segmentation, that is, the segmentation of text without blanks into words (or sometimes morphemes). For example, consider the Chinese and Japanese languages where text is written without delimiters between words. A first necessary task in the processing of these languages is to determine the probable locations of boundaries between words.

In the following, we will discuss a few aspects related to morphology learning and word segmentation. The existing algorithms in these fields include examples from both the supervised and unsupervised machine learning paradigms. We will focus on unsupervised and minimally supervised methods. For a broader overview, which includes work on supervised algorithms, the reader is referred to, for example, Goldsmith [2001] and Kit et al. [2002].

2.1 Challenges for Highly Inflecting and Compounding Languages

It is common that algorithms designed for morphology learning not only produce a segmentation of words into morphemes, but additionally attempt to discover relationships between words, such as knowledge of which word forms belong to the same inflectional paradigm. These higher-reaching goals are achieved by severely constraining the model space: prior assumptions regarding the inner structure of words (morphotactics) are expressed as strict constraints. Typically, words are restricted to consist of one stem followed by one, possibly empty, suffix as in, for example, Déjean [1998] and Snover and Brent [2001]. Goldsmith [2001] induces paradigms that he calls signatures. In doing this, he also proposes a recursive structure in which stems can consist of a substem and a suffix. Prefixes are also possible in Goldsmith's model.

In word segmentation such constraints are inapplicable because the number of words per sentence can vary greatly and is rarely known in advance. Commonly, algorithms designed for word segmentation utilize very little prior knowledge or assumptions about the syntax of the language. Instead, prior knowledge about typical word length may be applied, and small seed lexicons are sometimes used for bootstrapping. The segmentation algorithms try to identify character sequences that are likely words without consideration of the context in which the words occur (e.g., Ando and Lee [2000]; Yu [2000]; and Peng and Schuurmans [2001]).

For highly-inflecting and compounding languages, such as Finnish, both the outlined approaches are problematic. Typically word segmentation algorithms perform on an insufficient level, apparently due to the lack of any notion of morphotactics. On the other hand, typical morphology learning algorithms have problems because the ingrained assumptions they make about word structure are generally wrong (i.e., too strict) for Finnish or for other highly inflecting or compounding languages. In short, they cannot handle the possibly high number of morphemes per word. A Finnish word can consist of lengthy sequences of alternating stems and suffixes as in the example in Figure 1. Our attempts at finding a solution to this problem are described in the current article. Subsets of these results have previously been presented in the articles Creutz and

elämä	n	tapa	muutoks	i	lla
life	of	style	change	-s	with

Fig. 1. Morpheme segmentation of the Finnish word elämäntapamuutoksilla (with [the] changes of life style.)

Lagus [2002]; Creutz [2003]; and Creutz and Lagus [2004, 2005a]. However, the generalized structure and discussion on its components are presented here for the first time.

2.2 General Modeling Methodologies

There exist some central mathematical frameworks, or modeling methodologies, that can be used for formulating models for morphology learning and word segmentation.

In maximum likelihood (ML) modeling, only the accuracy of the representation of the data is considered when choosing a model, that is, model complexity (size of the model) is not taken into account. ML is known to lead to overlearning unless some restrictive model search heuristics or model smoothing is applied. There exist word segmentation and morphology learning algorithms where the complexity of the model is controlled heuristically, for instance, Ge et al. [1999], Peng and Schuurmans [2001], Kneissler and Klakow [2001], Creutz and Lagus [2004].

Probabilistic maximum a posteriori (MAP) models and equivalency, models based on the Minimum Description Length (MDL) principle, choose the best model by simultaneously considering model accuracy and model complexity; simpler models are favored over complex ones. This generally improves generalization capacity by inhibiting overlearning. A number of word segmentation and morphology learning algorithms have been formulated either using MDL or MAP, for example, de Marcken [1996], Deligne and Bimbot [1997], Kazakov [1997], Brent [1999], Kit and Wilks [1999], Yu [2000], Goldsmith [2001], Snover and Brent [2001], Creutz and Lagus [2002], and Creutz [2003]. In these works, the goal is to find the most likely lexicon (model) as well as a likely segmentation of the data. A more elaborate, and a much more computationally intensive, way of performing the task would be to use Bayesian model averaging. Instead of choosing one particular model, every possible model among some parameterized set is chosen with a weight that is proportional to the probability of the particular model. However, we are unaware of attempts to use such an approach in this task.

Finite state automata (FSA) can be used to describe the possible word forms of a language, for example, in the two-level morphology framework [Koskenniemi 1983]. There exist algorithms that try to learn FSA:s that compactly model the word forms observed in the training data [Johnson and Martin 2003; Goldsmith and Hu 2004]. Also Altun and Johnson [2001] induce a stochastic finite state automaton describing Turkish morphology, but their method works only in a supervised learning task, that is, they require a segmented, labeled corpus to begin with.

Parallels from the automaton approach can be drawn to methods, inspired by the works of Zellig S. Harris [1955, 1967] where a word or morpheme boundary is suggested at locations where the predictability of the next letter in a letter sequence is low, for instance, Déjean [1998], Ando and Lee [2000], Adda-Decker [2003], and Feng et al. [2004]. If the letter sequences (words or sentences) are sorted into a suffix tree, these low-predictability locations correspond to nodes with a high branching factor. The suffix tree could be compressed by merging nodes that have identical continuations, thereby producing a more compact data structure, which is a FSA.

2.3 Learning Morphological Structure

The model presented in this work provides a good means for the segmentation of words into morphemes. Alternatively, the model can be applied to word form generation. The rather few restrictions incorporated in the current model makes it a very permissive model of morphology. Such a model predicts a large number of words outside of the observed training corpus. This is desirable behavior since a successful learning algorithm should be able to generalize to unseen data. However, a permissive model also makes many mistakes. Many alternative approaches to morphology learning focus on the acquisition of more restrictive morphologies where far fewer words outside of the training corpus are recognized.

Some works discover pairs of related words or pairs of multiword collocations. Jacquemin [1997] discovers morphological variants of multiword collocations, for example, 'longitudinal record*ing*' vs. 'longitudinal*ly* record*ed*'. The collocations essentially have the same semantics and can be identified through regular suffix patterns, {e.g., (ϵ, ing), (ly, ed)}. Baroni et al. [2002] and Neuvel and Fulop [2002] propose algorithms that learn similarities in the spelling of word pairs. The discovery of patterns is not restricted to concatenation, but also include, for instance, vowel change such as the German Umlaut: Anschlag vs. Anschläge. Generation takes place by predicting missing word pairs. For instance, the pair receive vs. reception yields the pair deceive vs. deception by analogy (where it is assumed that the word deception was not in the training set).

Other works aim at forming larger groups of related word forms. Gaussier [1999] learns derivational morphology from inflectional lexicons. Orthographically similar words are clustered into relational families. From the induced word families, derivational rules can be acquired, such as the following French verb-to-noun conversions: produire $\rightarrow$ production, produire $\rightarrow$ producteur. Schone and Jurafsky [2000, 2001] make use of a Harris-like algorithm to separate suffixes and prefixes from word stems. Whether two orthographically similar word forms are morphologically related is determined from their context of neighboring words. A semantic representation for a word is obtained from the context using Latent Semantic Analysis (LSA). The semantic properties of a word are assumed to emerge from a large context window, whereas syntactic properties can be determined from a narrow window of the immediate word context. In addition to orthographic, semantic, and syntactic similarity, transitive closure is utilized as a forth component, that is, if conductive is related to conduct and conductivity is related to conductive, then conductivity is related to conduct.

Yarowsky and Wicentowski [2000] and Yarowsky et al. [2001] discover shared root forms for a group of inflected words. Verbs in numerous languages are studied. Frequency distributions are included as a clue to whether words are related. For instance, the English word singed can be discarded as a past tense candidate of 'to sing' because singed is far too rare. Furthermore, parallel corpora in multiple languages are utilized, and one language can function as a bridge for another language. For example, the French verb croire can be discovered as the root of croyaient, since these two forms are linked to the English verb believe in a parallel text. A missing link from the resembling verb forms croissant and croître tells us that these are not likely to be related to croîre. Wicentowski [2004] learns a set of string transductions from inflection root pairs and uses these to transform unseen inflections to their corresponding root forms. This model, however, is trained in a supervised manner.

A further step consists in inducing complete inflectional paradigms, that is, discovering sets of stems that can be combined with a particular set of suffixes. Goldsmith [2001] formulates his well-known algorithm Linguistica in an MDL framework, whereas Snover and Brent [2001] and Snover et al. [2002] present a similar, probabilistically formulated, model. These models do not predict any word forms outside of the training data. If the following English verb forms have been observed: talk, talks, talking, walk, walked, walks, the verbs talk and walk will go into separate paradigms, talk with the suffix set $\{\epsilon$, s, ing$\}$ and walk with the suffix set $\{\epsilon$, ed, s$\}$. More general paradigms can be obtained by collapsing them together, that is clustering them based on context similarity [Hu et al. 2005a]. This model can, in principle, predict the missing verb forms talked and walking.

As mentioned in Section 2.1, existing models make the learning of higher-level morphological structure computationally feasible by assuming that a word consists of maximally two, or three, morphemes. In recent work, Goldsmith and Hu [2004] and Hu et al. [2005b] move towards morphologies with a larger number of morphemes per word. A heuristic is described that is capable of learning 3- and 4-state FSA:s that model word forming in Swahili, a language with rich prefixation.

2.4 Composition of Meaning and Form

A central question regarding morpheme segmentation is the *compositionality* of meaning and form. If the meaning of a word is transparent in the sense that it is the sum of the meaning of the parts, then the word can be split into the parts, which are the morphemes, for example, English foot+print, joy+ful+ness, play+er+s. However, it is not uncommon that the form consists of several morphemes, which are the smallest elements of syntax, but the meaning is not entirely compositional, for example, English foot+man (male servant wearing a uniform), joy+stick (control device), sky+scrap+er (very tall building).

de Marcken [1996] proposes a model for unsupervised language acquisition, in which he defines two central concepts: *composition* and *perturbation*. Composition means that an entry in the lexicon is composed of other entries, for instance, joystick is composed of joy and stick. Perturbation means that changes are introduced that give the whole a unique identity, for example, the meaning of joystick is not exactly the result of the composition of the parts. This framework is similar to the class hierarchy of many programming languages where classes can modify default behaviors that are inherited from superclasses. The more of its properties a lexical parameter inherits from its components, the fewer need to be specified via perturbations.

Among other things, de Marcken [1996] applies his model in a task of unsupervised word segmentation of a text, where the blanks have been removed. As a result, hierarchical segmentations are obtained, for instance, for the phrase "for the purpose of": [[f[or]][[t[he]][[[p[ur]][[[po]s] e]][of]]]]. The problem here from a practical point of view is that there is no way of determining which level of segmentation corresponds best to a conventional word segmentation. On the coarsest level, the phrase works as an independent word (forthepurposeof). On the most detailed level, the phrase is shattered into individual letters.

3. FORMULATION OF THE MORFESSOR MODEL STRUCTURE

The determination of a suitable model family, that is, model structure, is of central importance since it sets a hard constraint on what can be learned in principle. A too restricting model family may exclude all optimal and near-optimal models, making learning a good model impossible, regardless of how much data and computation time is spent. In contrast, a too flexible model family is very hard to learn as it requires impractical amounts of data and computation.

We present Morfessor, a probabilistic model family for morphology learning. The model family consists of a number of distinct components which can be interpreted to encode both syntactic and semantic aspects of morphs, which are word segments discovered from data. Morfessor is a unifying framework that encompasses the particular models introduced earlier in Creutz and Lagus [2002], Creutz [2003], and Creutz and Lagus [2004, 2005a] and also has close connections to models proposed by other researchers. Each of these particular works has brought additional understanding regarding relevant problems and how they can be solved.

This section contains the mathematical formulation of the general model structure along with a discussion of the interpretation of its components. In Section 4, we outline how our earlier models can be seen as particular instances or subsets of this model. For a discussion on how to estimate any of the models (i.e., for the details of the model search algorithms), the interested reader is referred to our earlier publications.

3.1 Maximum a Posteriori Estimate of the Overall Probability

The task is to induce a model of language in an unsupervised manner from a corpus of raw text. The model of language ($\mathcal{M}$) consists of a morph vocabulary, or a lexicon of morphs, and a grammar. We aim at finding the optimal model of language for producing a segmentation of the corpus, that is, a set of morphs that is concise and, moreover, gives a concise representation for the corpus. The *maximum a posteriori* (MAP) estimate for the parameters, which is to be maximized, is:

$$\arg\max_{\mathcal{M}} P(\mathcal{M} \mid corpus) = \arg\max_{\mathcal{M}} P(corpus \mid \mathcal{M}) \cdot P(\mathcal{M}), \text{ where} \tag{1}$$
$$P(\mathcal{M}) = P(lexicon, grammar). \tag{2}$$

As can be seen from Equation (1), the MAP estimate consists of two parts: the probability of the model of language $P(\mathcal{M})$ and the maximum likelihood (ML) estimate of the corpus conditioned on the given model of language, written as $P(corpus \mid \mathcal{M})$. The probability of the model of language Equation (2) is the joint probability of the probability of the induced lexicon and grammar. It incorporates our assumptions of how some features should affect the morphology learning task. This is the Bayesian notion of probability, that is, using probabilities for expressing degrees of prior belief rather than counting relative frequency of occurrence in some empirical test setting.

In the following, we will describe the components of the Morfessor model in greater detail by studying the representation of the lexicon, grammar, and corpus, as well as their respective components.

3.2 Lexicon

The lexicon contains one entry for each distinct morph (morph type) in the segmented corpus. We use the term *lexicon* to refer to an inventory of whatever information one might want to store regarding a set of morphs, including their interrelations.

Suppose that the lexicon consists of M distinct morphs. The probability of coming up with a particular set of M morphs $\mu_1 \ldots \mu_M$ making up the lexicon can be written as:

$$P(lexicon) = P(size(lexicon) = M) \cdot P(properties(\mu_1), \ldots, properties(\mu_M)) \cdot M!. \tag{3}$$

The product contains three factors: (i) the prior probability that the lexicon contains exactly M distinct morphs, (ii) the joint probability that a set of M morphs, each with a particular set of properties, is created, and (iii) the factor $M!$, which is explained by the fact that there are $M!$ possible orderings of a set of M items and the lexicon is the same regardless of the order in which the M morphs emerged. (It is always possible to rearrange the morphs afterwards into an unambiguously defined order such as alphabetical order.)

The effect of the first factor, $P(size(lexicon) = M)$, is negligible since, in the computation of a model involving thousands of morphs and their parameters, one single probability value is of no practical significance. Thus, we have not defined a prior distribution for $P(size(lexicon))$.[1]

The properties of a morph can be divided into information regarding (1) the usage and (2) the form of the morph:

$$P(properties(\mu_i)) = P(usage(\mu_i), form(\mu_i)). \tag{4}$$

In Section 3.5, we present a set of properties, each of which corresponds to a

[1]If one were to define a proper prior, one possible choice would be Rissanen's universal prior for positive numbers (see Equation (14)).

component of the model, and group them under the usage and form aspects. The purpose of this grouping is to facilitate the understanding of the model; the model itself would be equivalent without it.

3.3 Grammar

Grammar can be viewed to contain information about how language units can be combined. In this work we model a simple morphotactics, that is, word-internal syntax. Instead of estimating the structure of the grammar from data, we currently utilize a specific fixed structure. Therefore, we do not have to calculate the probability of the grammar as a whole, and $P(\mathcal{M})$ in Equation (2) reduces to $P(lexicon)$.

The fixed structure of the grammar is taken as the following: morphs are generated from a small number of categories, *prefix* (PRE), *stem* (STM), *suffix* (SUF), and *nonmorpheme* (NON), which will be described more thoroughly later. Between the categories there are transition probabilities that exhibit the first-order Markov property. Words can consist of any number of morphs, which can be tagged with any categories, with a few restrictions. Suffixes are not allowed in the beginning and prefixes at the end of words; furthermore, it is impossible to move directly from a prefix to a suffix without passing through another morph.

It is possible for a morph to be assigned different categories in different contexts. The tendency of a morph μ_i to be assigned a particular category C_i, $P(C_i \mid \mu_i)$, (e.g., the probability that the English morph ness functions as a suffix) is derived from the parameters related to the usage of the morph:

$$P(C_i \mid \mu_i) = P(C_i \mid usage(\mu_i)). \tag{5}$$

The inverse probability, that is, the probability of a particular morph when the category is known, is needed for expressing the probability of the segmentation of the corpus. This emission probability $P(\mu_i \mid C_i)$ is obtained using Bayes' formula:

$$P(\mu_i \mid C_i) = \frac{P(C_i \mid \mu_i) \cdot P(\mu_i)}{P(C_i)} = \frac{P(C_i \mid \mu_i) \cdot P(\mu_i)}{\sum_{\forall \mu_{i'}} P(C_i \mid \mu_{i'}) \cdot P(\mu_{i'})}. \tag{6}$$

The category-independent probabilities $P(\mu_i)$ are maximum likelihood estimates, that is, they are computed as the frequency of the morph μ_i in the corpus divided by the total number of morph tokens.

3.4 Corpus

Every word form in the corpus can be represented as a sequence of some morphs that are present in the lexicon. Usually, there are many possible segmentations of a word. In MAP modeling, the one most probable segmentation is chosen. The probability of the corpus, when a particular model of language (lexicon and grammar) and morph segmentation is given, takes the form:

$$P(corpus \mid \mathcal{M}) = \prod_{j=1}^{W} \left[P(C_{j1} \mid C_{j0}) \prod_{k=1}^{n_j} \left[P(\mu_{jk} \mid C_{jk}) \cdot P\big(C_{j(k+1)} \mid C_{jk}\big) \right] \right]. \tag{7}$$

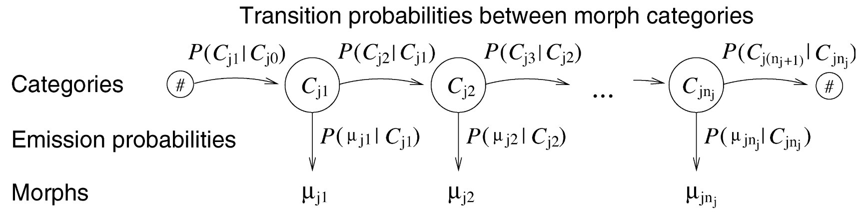


Fig. 2. The HMM model of a word according to Equation (7). The word consists of a sequence of morphs which are emitted from latent categories. For instance, a possible category sequence for the English word unavailable would be prefix + stem + suffix and the corresponding morphs would be un + avail + able.

As mentioned in Section 3.3, this is a Hidden Markov Model, and it is visualized in Figure 2. The product is taken over the W words in the corpus (token count), which are each split into n_j morphs. The kth morph in the jth word, μ_{jk}, is assigned a category, C_{jk}. The probability that the morph is emitted by the category is written as $P(\mu_{jk} \mid C_{jk})$. There are transition probabilities $P(C_{j(k+1)} \mid C_{jk})$ between the categories, where C_{jk} denotes the category assigned to the kth morph in the word, and $C_{j(k+1)}$ denotes the category assigned to the following, or $(k + 1)$th, morph. The transition probabilities comprise transitions from a special word boundary category (#) to the first morph in the word as well as the transition from the last morph to a word boundary.

3.5 Usage and Form of Morphs

In order to find general patterns of how a morph is used, information is collected about the distributional nature of the occurrences of the morph in the segmented corpus. We refer to this distribution as the usage of the morph. This includes both properties of the morph itself and properties of the context in which it typically appears. The typical usage of the morph can be parameterized and the parameters stored in the lexicon. Which parameter values are likely is determined by probability density functions (pdf:s), which are prior pdf:s in the Bayesian sense and favor solutions that are linguistically motivated. The features that have been used for modeling usage in this work, as well as possible extensions, are described in Section 3.5.2.

By the form of a morph we understand the symbolic representation of the morph, that is, the string of letters of which it consists. Different strings have different probabilities, which are determined using a prior probability distribution.

Given this distinction between usage and form, we make the assumption that they are statistically independent:

$$P(\mathit{properties}(\mu_1), \ldots, \mathit{properties}(\mu_M)) = \\ P(\mathit{usage}(\mu_1), \ldots, \mathit{usage}(\mu_M)) \cdot P(\mathit{form}(\mu_1), \ldots, \mathit{form}(\mu_M)). \tag{8}$$

3.5.1 *Form of a Morph.* In the current model, we further make the simplifying assumption that the forms of the morphs in the lexicon are independent of each other, thus

$$P(\mathit{form}(\mu_1), \ldots, \mathit{form}(\mu_M)) = \prod_{i=1}^{M} P(\mathit{form}(\mu_i)). \tag{9}$$

We draw inspiration from de Marcken [1996] in the sense that morphs in the lexicon have hierarchical structure. A morph can either consist of a string of letters or of two submorphs which can recursively consist of submorphs. The probability of the form of the morph μ_i depends on whether the morph is represented as a string of letters (Equation (10a)) or as the concatenation of two submorphs (Equation (10b)):

$$P(\mathit{form}(\mu_i)) = \begin{cases} (1 - P(sub)) \cdot \prod_{j=1}^{\mathit{length}(\mu_i)} P(c_{ij}). & \text{(10a)} \\ P(sub) \cdot P(C_{i1} \,|\, sub) \cdot P(\mu_{i1} \,|\, C_{i1}) \cdot P(C_{i2} \,|\, C_{i1}) \cdot P(\mu_{i2} \,|\, C_{i2}). & \text{(10b)} \end{cases}$$

$P(sub)$ is the probability that a morph has substructure, that is, the morph consists of two submorphs. $P(sub)$ is estimated from the lexicon by dividing the number of morphs having substructure by the total number of morphs.

In Equation (10a), $P(c_{ij})$ is the probability of the jth letter in the ith morph in the lexicon. The last letter of the morph is the *end-of-morph character* which terminates the morph. The probability distribution to use for the letters in the alphabet can be estimated from the corpus (or the lexicon).

Equation (10b) resembles Equation (7), where the probability of the corpus is given. $P(C_{i1} \,|\, sub)$ is the probability that the first morph in the substructure is assigned the category C_{i1}. $P(C_{i2} \,|\, C_{i1})$ is the transition probability between the categories of the first and second submorphs. $P(\mu_{i1} \,|\, C_{i1})$ and $P(\mu_{i2} \,|\, C_{i2})$ are the probabilities of the submorphs μ_{i1} and μ_{i2} conditioned on the categories C_{i1} and C_{i2}. The transition and morph emission probabilities are the same as in the probability of the corpus (Eq. 7). An example of concrete substructures are given later (Section 4.3, Figure 4).

3.5.2 *Features Related to the Usage of a Morph.* The set of features that could be used for describing usage is very large: The typical set of morphs that occur in the context of the target morph could be stored. Typical syntactic relations of the morph with other morphs could be included. The size of the context could vary from very limited to large and complex. A complex context might reveal different aspects of the usage of the morph from fine-grained syntactic categories to broader semantic, pragmatic, or topical distinctions. One might even use information from multimodal contexts (e.g., images, sounds) for grounding morph meaning to perceptions of the world. This reasoning relies on the philospohical view that the meaning of linguistic units (e.g., morphs) is reflected directly in how they are used.

However, in this work only a very limited set of features is used, based only on information contained in word lists. As properties of the morph itself, we count the *frequency* of the morph in the segmented corpus and the *length* in letters of the morph. As distilled properties of the context the morph occurs in, we consider the intraword *right* and *left perplexity*[2] of the morph.

Using these features, the probability of the usages of the morphs in the lexicon becomes:

$$P(usage(\mu_1), \ldots, usage(\mu_M)) =$$
$$P(freq(\mu_1), \ldots, freq(\mu_M)) \cdot \prod_{i=1}^{M} [P(length(\mu_i)) \cdot P(right\text{-}ppl(\mu_i)) \cdot P(left\text{-}ppl(\mu_i))]. \tag{11}$$

Due to practical considerations in the current implementation, we have assumed that the length and the right and left perplexity of a morph are independent of the corresponding values of other morphs. In contrast, the frequencies of the morphs are given as a joint probability, that is, there is one single probability for an entire morph frequency distribution. The probability distributions have been chosen due to their generality and simplicity. In a more sophisticated model formulation, one could attempt to model dependencies between morphs and their features, such as the general tendency of frequent morphs to be rather short.

Next, we describe the individual features and the prior probability distributions that are used for the range of possible values of these features. We conclude the treatment of morph usage by reporting how the usage of a morph translates into category membership probabilities in the current grammar. We stress that this particular grammar, as well as the set of features used, is only one possible solution among a large number of alternatives.

3.5.2.1 *Frequency.* Frequent and infrequent morphs generally have different semantics. Frequent morphs can be function words and affixes as well as common concepts. The meaning of frequent morphs is often ambiguous as opposed to rare morphs which are predominantly content words.

The knowledge of the frequency of a morph is required for calculating the value of $P(\mu_i)$ in Equation (6). The probability that a particular frequency distribution emerges is defined by the following prior probability:

$$P(freq(\mu_1), \ldots, freq(\mu_M)) = 1/\binom{N-1}{M-1} = \frac{(M-1)!(N-M)!}{(N-1)!}, \tag{12}$$

where N is the total number of morph tokens in the corpus, that equals the sum of the frequencies of the M morph types that make up the lexicon. Equation (12) is derived from combinatorics: as there are $\binom{N-1}{M-1}$ ways of choosing M positive integers that sum up to N, the probability of one particular frequency distribution of M frequencies summing to N is $1/\binom{N-1}{M-1}$.

3.5.2.2 *Length.* We assume that the length of a morph affects the probability of whether the morph is likely to be a stem or belong to another morph category. Stems often carry semantic (as opposed to syntactic) information. As the set of stems is very large in a language, stems are not likely to be very short morphs because they need to be distinguishable from each other.

[2]Perplexity, a function of entropy, describes how predictable the context is given this morph.

The length of a morph can be deduced from its form if an end-of-morph character is used (see Section 3.5.1). However, the consequence of such an approach is that the probability of observing a morph of a particular length decreases exponentially with the length of the morph which is clearly unrealistic. Instead of using an end-of-morph marker, one can explicitly model morph length with more realistic prior probability distributions. A Poisson distribution can be justified when modeling the length distributions of word and morph tokens, for instance, Nagata [1997], but for morph types (i.e., the set of morphs in the lexicon) a gamma distribution seems more appropriate [Creutz 2003].

$P(\mathit{length}(\mu_i))$ in Equation (11) assumes values from a gamma distribution if such is used as a prior for morph length. Otherwise, if morph length is modeled implicitly by using an end-of-morph marker, $P(\mathit{length}(\mu_i))$ is superfluous.

3.5.2.3 *Intraword Right and Left Perplexity.* The left and right perplexity give a very condensed image of the immediate context in which a morph typically occurs. Perplexity serves as a measure for the predictability of the preceding or following morph.

Grammatical affixes mainly carry syntactic information. They are likely to be common general purpose morphs that can be used in connection with a large number of other morphs. We assume that a morph is likely to be a prefix if it is difficult to predict what the following morph is going to be, that is, there are many possible right contexts of the morph and the right perplexity is high. Correspondingly, a morph is likely to be a suffix if it is difficult to predict what the preceding morph can be, and the left perplexity is high. The right perplexity of a target morph μ_i is calculated as:

$$\mathit{right\text{-}ppl}(\mu_i) = \left[\prod_{\nu_j \,\in\, \mathrm{right-of}(\mu_i)} P(\nu_j \mid \mu_i) \right]^{-\frac{1}{f_{\mu_i}}}. \tag{13}$$

There are f_{μ_i} occurrences of the target morph μ_i in the corpus. The morph tokens ν_j occur to the right of and immediately follow, the occurrences of μ_i. The probability distribution $P(\nu_j \mid \mu_i)$ is calculated over all such ν_j. Left perplexity can be computed analogously.[3]

As a reasonable probability distribution over the possible values of right and left perplexity, we use Rissanen's universal prior for positive numbers ([Rissanen 1989]):[4]

$$P(n) \approx 2^{-\log_2 c - \log_2 n - \log_2 \log_2 n - \log_2 \log_2 \log_2 n - \dots}, \tag{14}$$

where the sum includes all positive iterates, and c is a constant, about 2.865.

[3]In fact, the best results are obtained when only context morphs ν_j that are longer than three letters are included in the perplexity calculation. This means that the right and left perplexity are mainly estimates of the predictability of the stems that can occur in the context of a target morph. Including shorter morphs seems to make the estimates less reliable because of the existence of nonmorphemes (noise morphs).

[4]Actually Rissanen defines his universal prior over all nonnegative numbers, and he would write $P(n - 1)$ on the left side of the equation. Since the lowest possible perplexity is one, we do not include zero as a possible value in our formula.

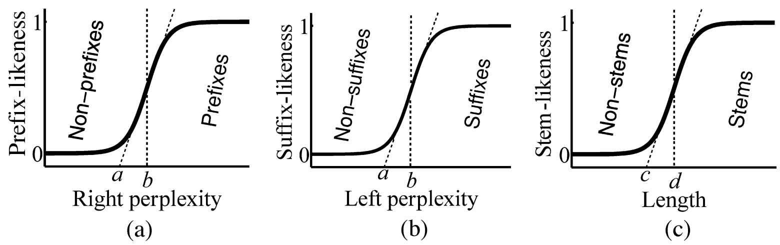


Fig. 3. Sketch of sigmoids, which express our prior belief of how the right and left perplexity as well as the length of a morph affects its tendency to function as a prefix, suffix, or stem.

To obtain $P(\textit{right-ppl}(\mu_i))$ and $P(\textit{left-ppl}(\mu_i))$, the variable n is substituted by the appropriate value, $\textit{right-ppl}(\mu_i)$ or $\textit{left-ppl}(\mu_i)$.

3.5.3 *Category Membership Probabilities.* In the grammar, the tendency of a morph to be assigned a particular category (PRE, STM, SUF, or NON) is determined by the usage (distributional nature) of the morph (Equation (5)). The exact relationship,

$$P(C_i \mid \textit{usage}(\mu_i)) = P(C_i \mid \textit{freq}(\mu_i), \textit{length}(\mu_i), \textit{right-ppl}(\mu_i), \textit{left-ppl}(\mu_i)), \quad (15)$$

could probably be learned purely from the data, but currently we use a fixed scheme involving a few adjustable parameters.

We obtain a measure of *prefix-likeness* by applying a graded threshold realized as a sigmoid function to the right perplexity of a morph (see Figure 3(a)):

$$\textit{prefix-like}(\mu_i) = (1 + \exp[-a \cdot (\textit{right-ppl}(\mu_i) - b)])^{-1}. \quad (16)$$

The parameter b is the perplexity threshold, which indicates the point where a morph μ_i is as likely to be a prefix as a nonprefix. The parameter a governs the steepness of the sigmoid. The equation for *suffix-likeness* is identical except that left perplexity is applied instead of right perplexity (Figure 3(b)).

As for stems, we assume that the *stem-likeness* of a morph correlates positively with the length in letters of the morph. A sigmoid function is employed, which yields:

$$\textit{stem-like}(\mu_i) = (1 + \exp[-c \cdot (\textit{length}(\mu_i) - d)])^{-1}. \quad (17)$$

where d is the length threshold and c governs the steepness of the curve (Figure 3(c)).

Prefix-, suffix- and stem-likeness assume values between zero and one, but they are not probabilities since they usually do not sum up to one. A proper probability distribution is obtained by first introducing the *nonmorpheme* category, which corresponds to cases where none of the proper morph classes is likely. Nonmorphemes are typically short, like the affixes, but their right and left perplexity are low, which indicates that they do not occur in a sufficient number of different contexts in order to qualify as a prefix or suffix. The probability that a segment is a nonmorpheme (NON) is:

$$P(\text{NON} \mid \mu_i) = [1 - \textit{prefix-like}(\mu_i)] \cdot [1 - \textit{suffix-like}(\mu_i)] \cdot [1 - \textit{stem-like}(\mu_i)]. \quad (18)$$

Then the remaining probability mass is distributed between prefix, stem, and suffix, for example,:

$$P(\text{PRE} \mid \mu_i) = \frac{\textit{prefix-like}(\mu_i)^q \cdot [1 - P(\text{NON} \mid \mu_i)]}{\textit{prefix-like}(\mu_i)^q + \textit{stem-like}(\mu_i)^q + \textit{suffix-like}(\mu_i)^q}. \tag{19}$$

The exponent q affects the normalization. High values of q produce spiky distributions (winner-take-all effect), whereas low values produce flatter distributions. We have tested the values $q = 1$ and $q = 2$.

As mentioned in Section 3.5.2.1, the frequency of a morph could possibly be used for distinguishing between semantic morphs (stems) and grammatical morphs (affixes). In the current scheme, the frequency as such is only used for computing the category-independent probabilities $P(\mu_i)$ (Eq. 6). Nonetheless, right and left perplexity are indirect measures of frequency, because a high frequency is a precondition for a high perplexity.

There is a similar idea of using the features frequency, mutual information, and left and right entropy in the induction of a Chinese dictionary from an untagged text corpus [Chang et al. 1995]. There the features are applied in classifying character sequences as either words or nonwords, which resembles our morpheme categories and the nonmorpheme category. In another work, Feng et al. [2004], a somewhat simpler feature called accessor variety was used in order to discover words in Chinese text. These features are not new within the field of word segmentation. In the pioneering work of Harris [1955] something very akin to accessor variety was introduced. Entropy was explored in a Harrisian approach to the segmentation of English words by Hafer and Weiss [1974]. However, in Morfessor, perplexity is not utilized to discover potential morph boundaries, but to assign potential grammatical categories to suggested morphs.

4. MODEL VARIANTS

Our earlier work can be seen as instances of the general Morfessor model since each of the previous models implements a subset of the components of Morfessor. These models and their central properties are summarized in Table I.

The widely known benchmark, John Goldsmith's algorithm Linguistica [Goldsmith 2001; Goldsmith 2005], is also included in the comparison even though it does not fit entirely into the Morfessor model family.

4.1 Baseline and Baseline-Length

The *Morfessor Baseline* model was originally presented as the Recursive MDL Model in Creutz and Lagus [2002]. The formulation followed from the Minimum Description Length (MDL) principle in a mathematically simplified way. In the Baseline, no context sensitivity is modeled, which corresponds to having only one morph category in the HMM in the grammar. The only feature related to morph usage that is taken into account is morph frequency. The form of the morph is flat, which means that a morph always consists of a string of letters and never has substructure. The Baseline model can be trained on a collection of either *word tokens* or *word types*. The former corresponds to a *corpus*, a

Table I. Summary of Some Properties of the Morphology Learning Algorithms
In the Optim. column, the nature of the optimization task is indicated. Context sensitivity (Ctxt-sens.) implies that the position in a word affects the probability of a morph, that is, some notion of morphotactics is included in the model. The Usage column lists the features of morph usage that are accounted for in the form of explicit prior distributions in the probability of the lexicon: frequency (F), gamma distribution for morph length (G), right (R) and left (L) perplexity. The structure of the form of morphs is given in the Form column. The Train column tells whether the model is trained on a corpus (tok: word token collection) or corpus vocabulary (typ: word type collection). The Long seq. column signals whether the model in question is suitable for morphologies where words can consist of lengthy sequences of morphemes.

Model Name	Optim.	Ctxt-sens.	Usage	Form	Train	Long seq.
Baseline	MAP	no	F	flat	tok & typ	yes
Baseline-Length	MAP	no	FG	flat	tok & typ	yes
Categories-ML	ML	yes	FGRL	flat	typ	yes
Categories-MAP	MAP	yes	FGRL	hierar.	tok (& typ)	yes
Linguistica	MAP	yes	–	signat.	typ (& tok?)	no

piece of text, where words can occur many times. The latter corresponds to a *corpus vocabulary* where only one occurrence of every distinct word form in the corpus has been listed. These two different types of data lead to different morph segmentations.

The choice of the term baseline signals that this model is indeed very simple. In essence, words are split into strings that occur frequently within the words in the corpus without consideration of the intraword context in which these segments occur. The more elaborate category-based Morfessor models make use of the Baseline algorithm in order to produce an initial segmentation which is then refined.

Morfessor Baseline-Length is a slight modification of the model introduced in Creutz [2003]. It is identical to the Baseline except that a gamma distribution is utilized for modeling morph length. Compared to the Baseline, the Baseline-Length algorithm performs better in a morpheme segmentation task, especially on small amounts of data, but the difference diminishes when the amount of data is increased.

Software implementing the Morfessor Baseline model variants is publicly available[5] under the GNU General Public License. User's instructions are provided in a technical report [Creutz and Lagus 2005b] which further describes the models and the search algorithm used. In brief, the search takes place as follows: the word forms in the corpus are processed, one at a time. First, the word as a whole is considered as a morph to be added to the lexicon. Then, every possible split of the word into two substrings is evaluated. The split (or no split) yielding the highest probability is selected. In case of a split, splitting of the two parts continues recursively and stops when no more gains can be obtained. All words in the corpus are reprocessed until convergence of the overall probability.

The advancement of the search algorithm can be characterized as follows. In order to split a word into two parts, the algorithm must recognize at least one of the parts as a morph. Initially, all entire word forms are considered potential

[5] `http://www.cis.hut.fi/projects/morpho/`.

morphs. Since many word stems occur in isolation as entire words (e.g., English match), the algorithm begins to discover suffixes and prefixes by splitting off the known stems from longer words (e.g., match+es, match+ing, mis+match). The newly discovered morphs can in turn be found in words where none of the parts occur in isolation (e.g., invit+ing). As a result of iterating this top-down splitting, the words in the corpus are gradually split down into shorter morphs.[6]

Both Baselines produce segmentations that are closer to a linguistic morpheme segmentation when trained on a word type collection instead of a word token collection. The use of word types means that all information about word frequency in the corpus is lost. If we are interested in drawing parallels to language processing in humans, this is an undesirable property because word frequency seems to play an important role in human language processing. Baayen and Schreuder [2000] refer to numerous psycholinguistic studies that report that high-frequency words are responded to more quickly and accurately than low-frequency words in various experimental tasks. This effect is obtained regardless of whether the words have compositional structure or not (both for regular derived and inflected words). Note, however, that these findings may not apply to all linguistic tasks. When test persons were exposed to word forms that were ungrammatical in context, high-frequency regular word forms seemed to be processed as if they were compositional rather than unanalyzed wholes [Allen et al. 2003].

4.2 Categories-ML

The *Morfessor Categories-ML* model has been presented in Creutz and Lagus [2004]. The model is a maximum likelihood (ML) model that is applied for reanalyzing a segmentation produced by the Baseline-Length algorithm. The morphotactics of the full Morfessor model is used in Categories-ML and all four usage features are included. However, the computation of the category membership probabilities (Section 3.5.3) is only utilized for initializing a category tagging of the morph segmentation obtained from Baseline-Length. Emission probabilities (Equation (6)) are then obtained as maximum likelihood estimates from the tagging.

The size of the morph lexicon is not taken into account directly in the calculation of the overall probability, but some heuristics are applied. If a morph in the lexicon consists of other morphs that are present in the lexicon (e.g., seemed = seem+ed), the most probable split (essentially according to Equation (10b)) is selected and the redundant morph is removed. A split into nonmorphemes is not allowed, however. If, on the contrary, a word has been shattered into many short fragments, these are removed by joining them with their neighboring morphs which hopefully creates a proper morph (e.g., flu+s+ter+ed becomes fluster+ed). This takes place by joining together nonmorphemes with their shortest neighbors until the resulting morph can qualify as a stem which is determined by

[6]Other search strategies could be explored in the future, especially when dealing with languages where free stems are rare, such as Latin (e.g., absurd+us, absurd+a, absurd+um, absurd+ae, absurd+o, etc.). However, initial experiments on Latin suggest that here also the current search algorithm manages to get a grip on the affixes and stems as the result of a long chain reaction.

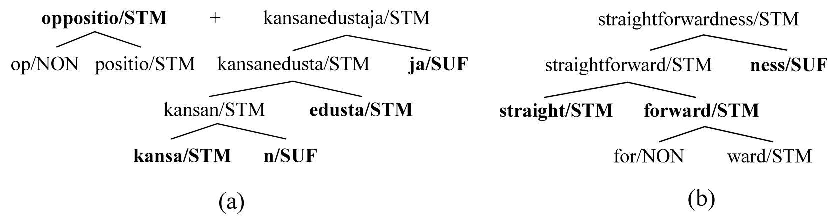


Fig. 4. The hierarchical segmentations of (a) the Finnish word oppositiokansanedustaja (MP of the opposition) and (b) the English word straightforwardness (obtained by the Categories-MAP model for the largest datasets). The finest resolution that does not contain nonmorphemes has been identified with boldface.

Equation (17). The Categories-ML algorithm operates on data consisting of word types.

4.3 Categories-MAP

The latest model, *Categories-MAP*, was introduced in Creutz and Lagus [2005a]. It is the most extensive model, and its formulation is the complete structure presented in Section 3.

The search for the most probable Categories-MAP segmentation takes place using a greedy search algorithm. In an attempt to avoid local maxima of the overall probability function, steps of resplitting and rejoining morphs are alternated; see Creutz and Lagus [2005a] for details including (i) initialization of a segmentation using Baseline-Length, (ii) splitting of morphs, (iii) joining of morphs using a bottom-up strategy, (iv) resplitting of morphs, (v) resegmentation of the corpus using the Viterbi algorithm and re-estimation of probabilities until convergence, (vi) repetition of Steps (iii)–(v) once.

Figure 4 shows hierarchical representations obtained by Categories-MAP for the Finnish word oppositiokansanedustaja (member of parliament of the opposition) and the English word straightforwardness. The Categories-MAP model utilizes information about word frequency: The English word has been frequent enough in the corpus to be included in the lexicon as an entry of its own. The Finnish word has been less common and is split into oppositio (opposition) and kansanedustaja (member of parliament) which are two separate entries in the lexicon induced from the Finnish corpus. Frequent words and word segments can thus be accessed directly which is economical and fast. At the same time, the inner structure of the words is retained in the lexicon because the morphs are represented as the concatenation of other (sub)morphs, which are also present in the lexicon: the Finnish word can be bracketed as [op positio][[[kansa n] edusta] ja] and the English word as [[straight [for ward]] ness].

Additionally, every morph is tagged with a category which is the most likely category for that morph in that context. Not all morphs in the lexicon need to be morpheme-like in the sense that they represent a meaning. Some morphs correspond more closely to syllables and other short fragments of words. The existence of these nonmorphemes (NON) makes it possible to represent some longer morphs more economically, e.g., the Finnish oppositio consists of op and positio (position), where op has been tagged as a nonmorpheme and positio as a stem. Sometimes this helps against the oversegmentation of rather rare words. When, for instance, a new name must be memorized, it can be constructed from shorter familiar fragments. This means that a fewer number of observations of this name in the corpus suffice for the name to be added as a morph to the lexicon compared to a situation where the name would need to be memorized letter-by-letter. For instance, in one of the English experiments the name Zubovski occurred twice in the corpus and was added to the morph lexicon as zubov/STM + ski/NON. One might draw a parallel from the nonmorphemes in the Categories-MAP model to findings within psycholinguistic research. McKinnon et al. [2003] suggest that morphological decomposition and representation extend to nonproductive morphemes, such as -ceive, -mit, and -cede in English words, e.g., conceive, permit, recede.

4.3.1 *Using Categories-MAP in a Morpheme Segmentation Task.* In the task of morpheme segmentation, the described data structure is very useful. While de Marcken (Section 2.4) had no means of knowing which level of segmentation is the desired one, we can expand the hierarchical representation to the finest resolution that does not contain nonmorphemes. In Figure 4, this level has been indicated using a boldface font. The Finnish word is expanded to oppositio+kansa+n+edusta+ja (literally opposition+people+of+represent+ative). The English word is expanded into straight + forward + ness. The morph forward is not expanded into for + ward, although this might be appropriate because for is tagged as a nonmorpheme in the current context.

4.4 Linguistica

The model of Linguistica is formulated in an MDL framework that is equivalent to a MAP model. In the Linguistica algorithm, a morphotactics is implemented where words are assumed to consist of a stem, optionally preceded by a prefix, and usually followed by a suffix. The stem can recursively consist of a substem and a succeeding suffix. This structure is less general than the one used in Morfessor because Linguistica does not allow consecutive stems (as in, e.g., coast+guard+s+man). Thus, morphologies involving compounding cannot be modeled satisfactorily.

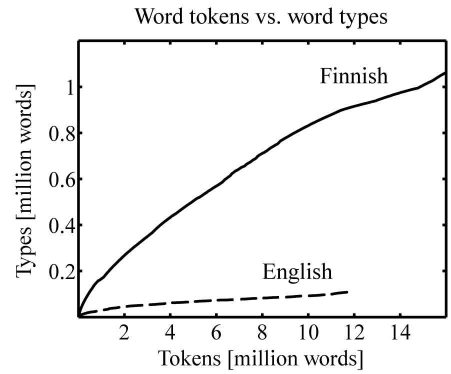
Linguistica groups stems and suffixes into collections called signatures (signat. in the Form column in Table I), which can be thought of as inflectional paradigms: a certain set of stems goes together with a certain set of suffixes. Words will be left unsplit unless the potential stem and suffix fit into a signature. Linguistica is trained on a word type collection, but it seems that word token collections could be used as well.

5. EXPERIMENTS

Careful evaluation of any proposed method is essential. Depending on the goal, the evaluation could be carried out directly in some NLP task such as speech recognition. However, as the performance in such a task depends on many issues and not only on the morphs, it is also useful to evaluate the morph segmentation directly.

word tokens	Finnish word types	English word types
10 000	5 500	2 400
50 000	20 000	7 200
250 000	65 000	17 000
12 000 000	—	110 000
16 000 000	1 100 000	—

(a)



(b)

Fig. 5. (a) Sizes of the test data subsets used in the evaluation. (b) Curves of the number of word types observed for growing portions of the Finnish and English test sets.

In the current article, the methods discussed are evaluated in a *linguistic morpheme segmentation task*. The goal is to find the locations of morpheme boundaries as accurately as possible. Experiments are performed on Finnish and English corpora and on datasets of different sizes. As a gold standard for the desired locations of the morpheme boundaries, *Hutmegs* is used (see Section 5.2). Hutmegs consists of fairly accurate conventional linguistic morpheme segmentations for a large number of word forms.

5.1 Finnish and English Datasets

The Finnish corpus consists of news texts from the CSC (The Finnish IT Center for Science)[7] and the Finnish News Agency (STT). The corpus contains 32 million words. It has been divided into a development set and a test set, each containing 16 million words.

For experiments on English, we use a collection of texts from the Gutenberg project (mostly novels and scientific articles),[8] and a sample from the Gigaword corpus and the Brown corpus.[9] The English corpus contains 24 million words. It has been divided into a development and a test set, each consisting of 12 million words. The development sets are utilized for optimizing the algorithms and for selecting parameter values. The test sets are used solely in the final evaluation.

What is often overlooked is that a comparison of different algorithms on one single dataset size does not give a reliable picture of how the algorithms behave when the amount of data changes. Therefore, we evaluate our algorithms with increasing amounts of test data. The amounts in each subset of the test set are shown in Figure 5(a), both as number of word tokens (words of running text) and number of word types (distinct word forms). Figure 5(b) further shows how

[7] `http://www.csc.fi/kielipankki/`.

[8] `http://www.gutenberg.org/browse/languages/en`.

[9] The Gigaword sample and the Brown corpus are available at the Linguistic Data Consortium: `http://www.ldc.upenn.edu/`.

the number of word types grows as a function of the number of word tokens for the Finnish and English test sets. As can be seen, for Finnish, the number of types grows fast when more text is added, that is, many new word forms are encountered. In contrast, with English text, a larger proportion of the words in the added text has been observed before.

5.2 Morphological Gold Standard Segmentation

The Helsinki University of Technology Morphological Evaluation Gold Standard (Hutmegs) [Creutz and Lindén 2004] contains morpheme segmentations for 1.4 million Finnish word forms and 120,000 English word forms. Hutmegs is based on the two-level morphological analyzer FINTWOL for Finnish [Koskenniemi 1983] and the CELEX database for English [Baayen et al. 1995]. These existing resources provide a morphological analysis of words but no surface-level segmentation. For instance, the English word bacteriologist yields the analysis bacterium+ology+ist. The main additional work related to the creation of Hutmegs consists in the semi-automatic production of surface-level, or allomorph, segmentations (e.g., bacteri+olog+ist). Hakulinen [1979] has been used as an authoritative guideline for the Finnish morphology and Quirk et al. [1985] for the English morphology. Both inflectional and derivational morphemes are marked in the gold standard.

The Hutmegs package is publicly available on the Internet.[10] For full access to the Finnish morpheme segmentations, an inexpensive license must additionally be purchased from Lingsoft, Inc.[11] Similarly, the English CELEX database is required for full access to the English material.[12]

As there can sometimes be many plausible segmentations of a word, Hutmegs provides several alternatives when appropriate, for instance, English evening (time of day) vs. even+ing (verb). There is also an option for so called fuzzy boundaries in the Hutmegs annotations which we have chosen to use. Fuzzy boundaries are applied in cases where it is inconvenient to define one exact transition point between two morphemes. For instance, in English, the stem-final e is dropped in some forms. Here we allow two correct segmentations, namely, the traditional linguistic segmentation in invite, invite+s, invit+ed and invit+ing, as well as the alternative interpretation, where the e is considered part of the suffix, as in: invit+e, invit+es, invit+ed and invit+ing.[13] In the former case, there are two allomorphs (realization variants) of the stem (invite and invit), and one allomorph for the suffixes. In the latter case, there is only one allomorph of the stem (invit), whereas there are two allomorphs of the third person present tense (-s and -es) and an additional infinitive ending (-e). Since there are a much greater number of different stems than suffixes in the English

[10] `http://www.cis.hut.fi/projects/morpho/`.
[11] `http://www.lingsoft.fi`.
[12] The CELEX databases for English, Dutch and German are available at the Linguistic Data Consortium: `http://www.ldc.upenn.edu/`.
[13] Note that the possible segmentation invite+d is not considered correct due to the fact that there is no indication that the regular past tense ending -ed ever loses its e, whereas the preceding stem unquestionably does so, for example, in inviting.

language, the latter interpretation lends itself to more compact concatenative models of morphology.

5.3 Evaluation Measures

As evaluation measures, we use *precision* and *recall* on discovered morpheme boundaries. Precision is the proportion of correctly discovered boundaries among all discovered boundaries by the algorithm. Recall is the proportion of correctly discovered boundaries among all correct boundaries. A high precision thus tells us that when a morpheme boundary is suggested, it is probably correct, but it does not tell us the proportion of missed boundaries. A high recall tells us that most of the desired boundaries were indeed discovered, but it does not tell us how many incorrect boundaries were suggested as well.

In order to get a comprehensive idea of the performance of a method, both measures must be taken into account. A measure that combines precision and recall is the *F-measure*, which is the harmonic mean of the two:

$$\textit{F-Measure} = 1/\left[\frac{1}{2}\left(\frac{1}{\textit{Precision}} + \frac{1}{\textit{Recall}}\right)\right]. \tag{20}$$

We compare performances using all three measures.

Furthermore, the evaluation measures can be computed either using word tokens or word types. If the segmentation of word tokens is evaluated, frequent word forms will dominate in the result because every occurrence (of identical segmentations) of a word is included. If, instead, the segmentation of word types is evaluated, every distinct word form, frequent or rare, will have equal weight. When inducing the morphology of a language, we consider all word forms to be important regardless of their frequency. Therefore, in this article, precision and recall for word types is reported.

For each of the data sizes 10,000, 50,000, and 250,000 words, the algorithms are run on five separate subsets of the test data, and the average results are reported. Furthermore, statistical significance of the differences in performance have been assessed using T-tests. The largest datasets, 16 million words (Finnish) and 12 million words (English), are exceptions since they contain all available test data which constrains the number of runs to one.

5.4 Methods to be Evaluated

We report experiments on the following methods from the Morfessor family: Baseline-Length, Categories-ML and Categories-MAP (see Table I for a concise description). The Baseline-Length model was trained on a collection of word types. Parameter values related to the priors of the category models (a, b, c, d, and q in Equations (16), (17) and (19)) were determined from the development set. The model evaluation was performed using independent test sets.

In addition, we benchmark against Linguistica [Goldsmith 2001, 2005].[14] In the Linguistica algorithm, we used the commands Find suffix system and

[14]We have used the December 2003 version of the Linguistica program that is publicly available on the Internet `http://humanities.uchicago.edu/faculty/goldsmith/Linguistica2000/`.

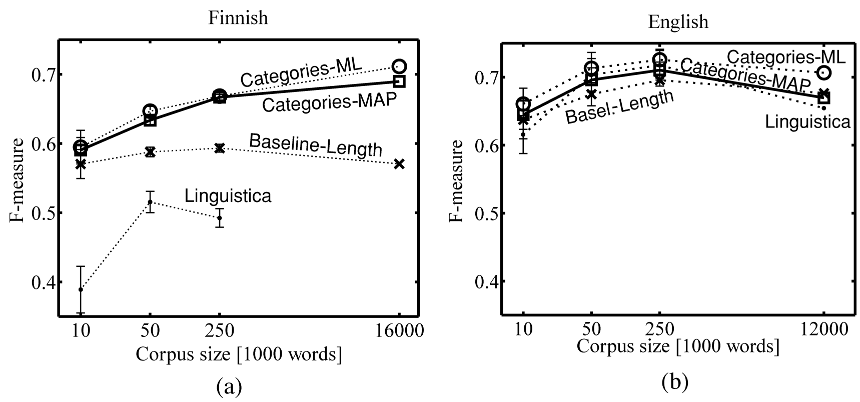


Fig. 6. Morpheme segmentation performance (F-measure of discovered morpheme boundaries) of the algorithms on (a) Finnish and (b) English test data. Each data point is an average of 5 runs on separate test sets with the exception of the 16 million words for Finnish and the 12 million words for English (1 test set). In these cases, the lack of test data constrained the number of runs. The standard deviations of the averages are shown as intervals around the data points. There is no data point for Linguistica on the largest Finnish test set because the program is unsuited for very large amounts of data due to its considerable memory consumption.

Find prefixes of suffixal stems. We interpreted the results in two ways: (i) to allow a word to be segmented into a maximum of three segments—an optional prefix, followed by a stem, followed by an optional suffix; (ii) to decompose stems that consist of a substem and a suffix which makes it possible for a word to be segmented into more than three segments. The former solution (i) surprisingly produced better results, and thus these results are reported in this work.

5.5 Results

Figures 6–8 depict the morph splitting performance of the evaluated methods in the Finnish and English morph segmentation tasks. The F-measures shown in Figure 6 allow for a direct comparison of the methods, whereas the precisions in Figure 7 and the recalls in Figures 8 shed more light on their relative strengths and weaknesses. Furthermore, some examples of the segmentations produced are listed in Tables II and III.

We will now briefly comment on the performance of each method in relation to the other methods.

5.5.1 *Baseline-Length.* When evaluated against a linguistic morpheme segmentation, the Baseline methods suffer because they undersegment frequent strings (e.g., English having, soldiers, states, seemed), especially when trained on word token collections (where several word forms occur a high number of times). With more data, the undersegmentation problem also becomes more severe when trained on word type collections (where each unique word form is encountered only once). This is due to the fact that the addition of more examples of frequent word segments justify their inclusion as morphs of their own in the lexicon. This shows as a decrease in overall performance on the largest data sizes in Figure 6 and in recall in Figure 8.

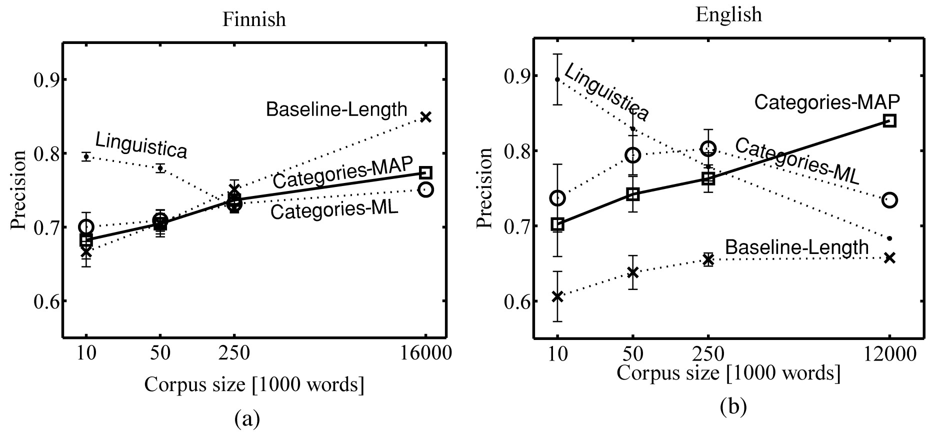


Fig. 7. Precision of discovered morpheme boundaries obtained by the algorithms on (a) Finnish and (b) English data. Standard deviations are shown as intervals around the data points.

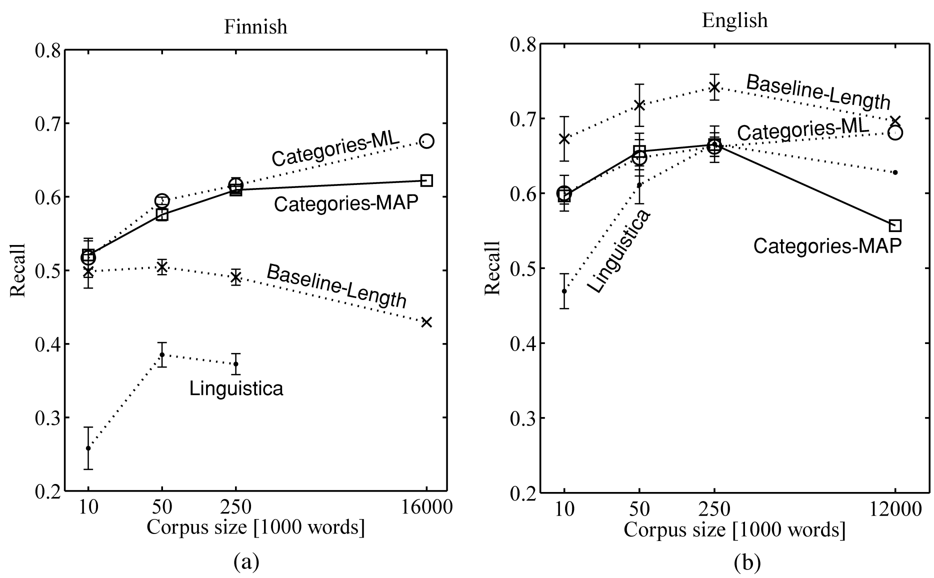


Fig. 8. Recall of discovered morpheme boundaries obtained by the algorithms on (a) Finnish and (b) English data. Standard deviations are shown as intervals around the data points. Precision (Figure 7) measures the accuracy of the proposed splitting points, while, recall describes the coverage of the splits.

The opposite problem consists in the oversegmentation of infrequent strings (e.g., flu+s+ter+ed). Moreover, the method makes segmentation errors that ensue from considering the goodness of each morph without looking at its context in the word, causing errors such as in Table II ja+n+ille where ja is incorrectly identified as a morph because it is frequently used as a suffix in the Finnish language. These kinds of segmentation errors are particularly common with English, which explains the generally low precision of the method in Figure 7(b).

Table II.
Examples of Finnish morpheme segmentations learned by versions of Morfessor from the 16 million word Finnish test set. Additionally, the corresponding gold standard segmentations are supplied. Proposed prefixes are underlined, stems are rendered in **bold-face**, and suffixes are *slanted*. Square brackets [] indicate higher-level stems and parentheses () higher-level suffixes in the hierarchical lexicon

Baseline-Length	Categories-ML	Categories-MAP	Gold standard
aarre kammioissa	**aarre kammio** *i ssa*	[**aarre kammio**] *issa*	**aarre kammio** *i ssa*
aarre kammioon	**aarre kammio** *on*	[**aarre kammio**] *on*	**aarre kammio** *on*
bahama laiset	**bahama lais** *et*	**bahama** *laiset*	**bahama** *laise t*
bahama saari en	**bahama saar** *i en*	**bahama** [**saari** *en*]	**bahama saar** *i en*
epä esteettis iksi	epä **este** *et t isi ksi*	epä [[**esteet** *ti*] *s*] *iksi*	**epäesteett** *is i ksi*
epätasapaino inen	epä tasa **paino** *in en*	[epä [[tasa **paino**] *inen*]]	epätasa **painoinen**
haapa koskeen	**haap** *a* **koske** *en*	[**haapa** [**koskee** *n*]]	**haapa koske** *en*
haapa koskella	**haap** *a* **koske** *lla*	[**haapa** [**koske** *lla*]]	**haapa koske** *lla*
ja n ille	**jani** *lle*	**jani** *lle*	**jani** *lle*
jäädyttä ä kseen	**jäädy** *ttä ä kseen*	[**jäädy ttää**] *kseen*	**jäädy** *ttä ä kse en*
ma clare n	**maclare** *n*	**maclare** *n*	–
nais autoilija a	nais **auto** *ili ja a*	[nais [**autoili** *ja*]] *a*	**nais autoili** *ja a*
pää aiheesta	pää **aihe** *e sta*	**pää** [**aihe** *esta*]	**pää aihee** *sta*
pää aiheista	pää **aihe** *i sta*	[pää [**aihe** *ista*]]	**pää aihe** *i sta*
päähän	**pää** *hän*	[pää **hän**]	**pää** *hän*
sano t takoon	**sano** *tta ko on*	[**sano ttakoon**]	**sano** *tta ko on*
sano ttiin ko	**sano** *tti in ko*	[**sano** *ttiin*] *ko*	**sano** *tt i in ko*
työ tapaaminen	työ **tapa** *a* **mine** *n*	**työ** [**tapaa** *minen*]	**työ tapaa** *minen*
töhri misistä	**töhri** *mis i stä*	**töhri** (*mis istä*)	**töhri** *mis i stä*
voi mmeko	**voim meko**	[[**voi** *mme*] *ko*]	**voi** *mme ko*
voisi mme kin	**voisi** *mme kin*	[**voisi** *mme*] *kin*	**vo** *isi mme kin*

Table III.
Examples of English morpheme segmentations learned by the four algorithms from the 12 million word English test set

Baseline-Length	Categories-ML	Categories-MAP	Linguistica
accomplish es	**accomplish** *es*	[**accomplish** *es*]	ac **compli** *shes*
accomplish ment	**accomplish** *ment*	[**accomplish ment**]	**accomplish** *ment*
beautiful ly	**beauti** *ful ly*	[**beautiful** *ly*]	**beautiful** *ly*
configu ration	con **figu** *r ation*	[**configur ation**]	con **figura** *tion*
dis appoint	dis **appoint**	**disappoint**	**disappoi** *nt*
expression istic	**express** *ion ist ic*	**expression istic**	**expression** *istic*
express ive ness	**express** *ive ness*	[**expressive** *ness*]	**expressive** *ness*
flu s ter ed	**fluster** *ed*	[**fluster** *ed*]	**fluster** *ed*
insur e	**insur** *e*	**insure**	**insur** *e*
insur ed	**insur** *ed*	[**insur** *ed*]	**insur** *ed*
insur es	**insur** *es*	[**insure** *s*]	**insure** *s*
insur ing	**insur** *ing*	[**insur** *ing*]	**insur** *ing*
long fellow 's	**long fellow** *'s*	[[**long fellow**] *'s*]	**longfellow** *'s*
master piece s	**master piece** *s*	[[**master piece**] *s*]	**masterpiece** *s*
micro organism s	**micro organ** *ism s*	[**micro** [**organism** *s*]]	micro **organism** *s*
photograph ers	**photo graph** *ers*	[[[**photo graph**] *er*] *s*]	**photograph** *ers*
present ly found	**present lyfound**	[**present** *ly*] **found**	**presentlyfou** *nd*
re side d	re **side** *d*	**resided**	**resid** *ed*
re side s	re **side** *s*	[**reside** *s*]	**reside** *s*
re s id ing	re **sid** *ing*	[re **siding**]	**resid** *ing*
un expect ed ly	un **expect** *ed ly*	[[un [**expect** *ed*]] *ly*]	un **expected** *ly*

5.5.2 *Categories-ML.* Out of the compared methods, Categories-ML consistently shows the highest results in Figure 6 for both Finnish and English with all data sizes. When compared to Baseline-Length in Figures 7(a) and 8(a), it appears that the considerable improvement is due to the fact that many previously undersegmented words have been split into smaller parts by Categories-ML. Many of the new proposed boundaries are correct (higher recall), but some are incorrect (lower precision). Apparently the simple morphotactics helps correct many mistakes caused by the lack of specific contextual information. However, the morphotactics is fairly primitive, and, consequently, new errors emerge when incorrect alternations of stems and suffixes are proposed, for example, Finnish, epä+este+et+t+isi+ksi (plural translative of epäesteettinen, unaesthetic), työ+tapa+a+mine+n (työ+tapaa+minen, job meeting). The drop in precision for Categories-ML with the largest English dataset (Figure 7(b)) is apparently caused by the multitude of word forms (many foreign words and names), which give rise to the discovery of suffixes that are not considered correct in contemporary English, for instance, plex+us, styl+us.

5.5.3 *Categories-MAP.* Figure 6(a) shows that Categories-MAP challenges Categories-ML as the best-performing algorithm for Finnish. For two data sizes (10,000 and 250,000 words), the difference between the two is not even statistically significant (T-test level 0.05). Also for English (Figure 6(b)) where the difference between all the algorithms is overall smaller than for Finnish, Categories-MAP places itself below the best-forming Categories-ML and above the Baseline-Length method except on the largest dataset where it falls slightly below Baseline-Length. Note, however, that the difference in performance is statistically significant only between Categories-ML and the lowest-scoring algorithm at each data size (Linguistica at 10,000 words; Baseline-Length at 50,000 and 250,000 words).

When looking at the detailed measures in Figures 7 and 8, one can see that Categories-MAP performs very well for Finnish, with both precision and recall rising as new data is added. However, for English, there is a fallback in recall on the largest dataset (Figure 8(b)), which is also reflected in decreased F-measure. This seems to be due to the fact that only the most frequent English prefixes and suffixes are detected reliably. In general, Categories-MAP is a more conservative splitter than Categories-ML.

5.5.4 *Linguistica.* Linguistica is a conservative word splitter for small amounts of data, which explains the low recall but high precision for small data sets. As the amount of data increases, recall goes up and precision goes down because more and more signatures (paradigms) are suggested, some of them correct and some incorrect. At some point, the new signatures proposed are mainly incorrect, which means that both precision and recall decrease. This can be observed as peculiar suffixes of words, for instance, disappoi+nt, longitu+de, presentlyfou+nd, sorr+ow. The recall of Linguistica can never rise very high because the algorithm only separates prefixes and suffixes from the stem and thereby misses many boundaries in compound words, for example, longfellow+'s, masterpiece+s, thanksgiv+ing.

Linguistica cannot compete with the other algorithms on the Finnish data, but, for English, it works at the level of Categories-ML for the datasets containing 50,000 and 250,000 words. (Note that Linguistica was not run on datasets for Finnish larger than 250,000 words because the program is unsuited for very large amounts of data due to its considerable memory consumption.)

5.5.5 *Behavior with Different Amounts of Data.* In the experiments on Finnish, Categories-ML and Categories-MAP both improve their performance with the addition of more data. The rising curves may be due to the fact that these models have more parameters to be estimated than the other models due to the HMM model for categories. The larger number of free parameters require more data in order to obtain good estimates. However, on the largest English set, all algorithms have difficulties, which seems to be due to the many foreign words contained in this set: patterns are discovered that do not belong to contemporary English morphology.

Linguistica does not benefit from increasing amounts of data. The best results were obtained with medium-sized datasets, around 50,000 words for Finnish and 250,000 words for English. Similarly, Baseline-Length does not seem to benefit from ever-increasing data sizes as it reaches its best performance with the data sets of 250,000 words.

5.6 Computational Requirements

The Categories-MAP algorithm was implemented as a number of Perl scripts and makefiles. The largest Finnish dataset took 34 hours and the largest English set $2\frac{1}{2}$ hours to run on an AMD Opteron 248, 2200MHz processor. The memory consumption never exceeded 1GB. The other algorithms were considerably faster (by an order of magnitude), but Linguistica was very memory-consuming.

6. DISCUSSION

Only the accuracy of the placement of morph boundaries has been evaluated quantitatively in the current article. It is worth remembering that the gold standard for splitting used in these evaluations is based on a traditional morphology. If the segmentations were evaluated using a real-world application, perhaps somewhat different segmentations would be most useful. For example, the tendency to keep common words together, seen in the Baseline and Categories-MAP models, might not be bad, for instance, in speech recognition or machine translation applications. In fact, quite the opposite, excessive splitting might be a problem in both applications.

The algorithms produce different amounts of information: the Baseline and Baseline-Length methods only produce a segmentation of the words, whereas the other algorithms (Categories-ML, Categories-MAP and Linguistica) also indicate whether a segment functions as a prefix, stem, or suffix. Additionally, by expanding the entries in the lexicon learned by Categories-MAP, a hierarchical representation is obtained, which can be visualized using a tree structure or nested brackets.

We will use the example segmentations obtained for a number of Finnish and English words (in Tables II and III) to briefly illustrate some aspects beyond the discovery of an accurate morpheme segmentation of words. In Table II, the gold standard segmentations for the Finnish words are given as a reference, whereas examples for Linguistica are lacking because the algorithm could not be run on the largest Finnish test set. English results are available for Linguistica in Table III, but the corresponding gold standard segmentations are not included due to limited space and to the fact that all readers are familiar with the English language. Readers interested in the analyses of further word forms can try our demonstration program on the Internet.[15]

6.1 Tagging of Categories

As has been shown, the introduction of a simple morphotactics, or word-internal syntax, in the Categories models reduced the occurrences of undersegmented and oversegmented words as well as misalignments due to the insensitivity of context which were observed in the Baseline models. Examples of such cases in Tables II and III comprise the Finnish words aarre+kammio+i+ssa (in treasure chambers), jani+lle (for Jani), sano+tta+ko+on (may it be said); and the English words photo+graph+er+s and fluster+ed.

Additionally, the simple morphotactics can sometimes resolve semantic ambiguities when the same morph is tagged differently in different contexts, for example, pää is a prefix in pääaiheesta and pääaiheista (about [the] main topic(s)), whereas pää is a stem in päähän (in [the] head). (In this example, the Categories-ML algorithm tagged the occurrences of pää correctly, while Categories-MAP made some mistakes.)

From the point of view of natural language processing, the identification and separation of semantic segments (mainly stems) and syntactic segments (mainly affixes) can be beneficial. The stems contained in a word form could be considered as a canonic (or base) form of the word, while the affixes could be considered as inflections. Such a canonic form for words could be an alternative to the base forms retrieved by handmade morphological analyzers or stemming algorithms which are used, for example, in information retrieval.

6.2 Bracketing

The hierarchical representation produced by the Categories-MAP algorithm, shown using nested brackets in Tables II and III, can be interpreted as the attachment hierarchy of the morphemes. With the current model, the construction of the hierarchy is likely to take place in the order of most frequently cooccurring word segments. Sometimes this is also grammatically elegant, for example, Finnish [epä [[tasa paino] inen]] (imbalanced, literaly bracketed as [un [[even weight] ed]]), [nais [autoili ja]] a (partitive of [female [car-driv er]]; English: [[[photo graph] er] s], [[un [expect ed]] ly]. But the probability of coming up with grammatically less elegant solutions is also high, for example, English [micro [organism s]]. (Note that the gold standard

[15] `http://www.cis.hut.fi/projects/morpho/`.

segmentation for epätasapainoinen is strange. Categories-MAP produces the correct segmentation.)

6.3 Overgeneralization

The algorithms can incorrectly overgeneralize and, for instance, suggest a suffix where there is none, such as, maclare+n (MacLaren). Furthermore, nonsensical sequences of suffixes (which in other contexts are true suffixes) can be suggested, such as, epä+este+et+t+isi+ ksi, which should be epä+esteett+is+i+ksi. A model with more fine-grained categories might reduce such shortcomings in that it could model morphotactics more accurately.

The use of signatures in Linguistica should conceivably prevent overgeneralization. In general, to propose the segmentation maclare+n, other forms of the proposed stem would be expected to occur in the data, such as maclare or maclare+ssa. If none of these exist, the segmentation should be discarded. However, especially with large amounts of data. Linguistica is oversensitive to common strings that occur at the end of words and proposes segmentations such as allu+de, alongsi+de, longitu+de; anyh+ow, highbr+ow, longfell+ow.

Solutions to this problem could be found, in an approach such as that advocated by Yarowsky and Wicentowski [2000], who study how distributional patterns in a corpus can be utilized to decide whether words are related or not. For instance, their method is able to determine that the English word singed is not an inflected form of to sing.

6.4 Allomorphy

Allomorphs are morphs representing the same morpheme, that is, morphs having the same meaning but used in complementary distributions. The current algorithms cannot in principle discover which morphs are allomorphs, for instance, that in Finnish on and en mark the same case, namely, illative, in aarre+kammio+on (into [the] treasure chamber) and Haapa+koske+en (to Haapakoski).[16] To enable such discovery in principle, one would probably need to look at contexts of nearby words, not just the word-internal context. Additionally, one should allow for the learning of a model with richer category structure.

Moreover, on and en do not always mark the illative case. In bahama+saari+en (of the Bahama islands), the genitive is marked as en, and in sano+tta+ko+on (may it be said), on marks the third person singular. Similar examples can be found for English such as, ed and d are allomorphs in insur+ed vs. re+side+d, and so are es and s in insur+es vs. re+side+s (Categories-ML).

Many cases of allomorphy can be modeled using morphophonological rules. The so-called Item and Process (IP) model of morphology assumes that some canonic forms of morphemes are appended to each other to form words and, when the morphemes meet, sound changes may typically ensue at the morpheme boundaries. For instance, the final e in insure is dropped when followed by the suffix ed. In principle, such rules could be learned in an unsupervised

[16]Furthermore the algorithm cannot deduce that the illative is actually realized as a vowel lengthening + n: kammioon vs. koskeen.

manner from unannotated data. Kontorovich et al. [2003] apply machine learning in the acquisition of allomorphic rules, but their method requires aligned training data.

Quite generally, much of the work in unsupervised morphology learning does not focus on concatenative morphology, that is, the discovery of consecutive word segments. Some algorithms learn relationships between words by comparing the orthographic and semantic similarity of pairs of words, (e.g., Neuvel and Fulop [2002] and Baroni et al. [2002]). These approaches can handle nonconcatenative morphological processes such as the unmlaut sound change in German. However, none of these models as such suits highly-inflecting languages as they assume only two or three constituents per word, analogous to possible prefix, stem, and suffix.

Moreover, there is some evidence that humans may simply memorize allomorphs without applying morphophonological transformations on what they hear (or read) [Järvikivi and Niemi 2002]. In this case, the morphology-learning models presented in this work are perhaps closer to human language processing than the IP model of morphology.

7. CONCLUSION

We have attempted to provide the reader with a broad understanding of the morphology learning problem. It is hoped that the general probabilistic model family presented, called Morfessor, and the discussion of each component opens new and fruitful ways to think about modeling morphology learning. The experimental comparison of different instances of the general model in a morpheme segmentation task sheds light on the usefulness and role of particular model components.

The development of good model search algorithms deserves additional consideration in the future. The categorial labelings of the morphs produced by the later model variants might be useful in other tasks such as information retrieval. An interesting avenue for future research is the consideration of how to extend the feature set applied in the modeling of morph usage, possibly to the point where one is able to ground meanings of morphs using multimodal information.

ACKNOWLEDGMENTS

We are grateful for inspirational discussions with Mikko Kurimo, Teemu Hirsimäki, Vesa Siivola, and Panu Somervuo from the Speech group of the NNRC. We thank Krister Lindén for enthusiastic discussions and valuable comments on the manuscript and Timo Honkela and Harri Valpola for inspirational support. Sami Virpioja deserves our special gratitude for the implementation and upgrades of the Internet demonstration of the method. We also thank the editor and the anonymous reviewers for their thorough work and valuable comments.

REFERENCES

ADDA-DECKER, M. 2003. A corpus-based decompounding algorithm for German lexical modeling in LVCSR. In *Proceedings of the 8th European Conference on Speech Communication and Technology (Eurospeech)*. Geneva, Switzerland. 257–260.

ALLEN, M., BADECKER, W., AND OSTERHOUT, L. 2003. Morphological analysis in sentence processing: An ERP study. *Lang. Cognit. Proc. 18*, 4, 405–430.

ALTUN, Y. AND JOHNSON, M. 2001. Inducing SFA with ϵ-transitions using Minimum Description Length. In *Proceedings of the Finite-State Methods in Natural Language Processing, ESSLLI Workshop*. Helsinki, Finland.

ANDO, R. K. AND LEE, L. 2000. Mostly-unsupervised statistical segmentation of Japanese: Applications to Kanji. In *Proceedings of the 6th Applied Natural Language Processing Conference and 1st Meeting of the North American Chapter of the Association for Computational Linguistics (ANLP-NAACL)*. 241–248.

BAAYEN, R. H., PIEPENBROCK, R., AND GULIKERS, L. 1995. The CELEX lexical database (CD-ROM). Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA. `http://wave.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC96L14`.

BAAYEN, R. H. AND SCHREUDER, R. 2000. Towards a psycholinguistic computational model for morphological parsing. *Philosophical Transactions of the Royal Society (Series A: Mathematical, Physical and Engineering Sciences 358)*, 1–13.

BARONI, M., MATIASEK, J., AND TROST, H. 2002. Unsupervised learning of morphologically related words based on orthographic and semantic similarity. In *Proceedings of the Workshop on Morphological & Phonological Learning of ACL*. 48–57.

BRENT, M. R. 1999. An efficient, probabilistically sound algorithm for segmentation and word discovery. *Machine Learn. 34*, 71–105.

CHANG, J.-S., LIN, Y.-C., AND SU, K.-Y. 1995. Automatic construction of a Chinese electronic dictionary. In *Proceedings of the 3rd Workshop on Very Large Corpora*. Somerset, NJ. 107–120.

CREUTZ, M. 2003. Unsupervised segmentation of words using prior distributions of morph length and frequency. In *Proceedings of the Association for Computations Languages* (ACL'03). Sapporo, Japan. 280–287.

CREUTZ, M. AND LAGUS, K. 2002. Unsupervised discovery of morphemes. In *Proceedings of the Workshop on Morphological and Phonological Learning of ACL*. Philadelphia, PA. 21–30.

CREUTZ, M. AND LAGUS, K. 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)*. Barcelona, Spain. 43–51.

CREUTZ, M. AND LAGUS, K. 2005a. Inducing the morphological lexicon of a natural language from unannotated text. In *Proceedings of the International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning (AKRR'05)*. Espoo, Finland. 106–113.

CREUTZ, M. AND LAGUS, K. 2005b. Unsupervised morpheme segmentation and morphology induction from text corpora using Morfessor 1.0. Tech. rep. A81, Publications in Computer and Information Science, Helsinki University of Technology.

CREUTZ, M. AND LINDÉN, K. 2004. Morpheme segmentation gold standards for Finnish and English. Tech. rep. A77, Publications in Computer and Information Science, Helsinki University of Technology.

DE MARCKEN, C. G. 1996. Unsupervised language acquisition. Ph.D. thesis, MIT, Cambridge, MA.

DÉJEAN, H. 1998. Morphemes as necessary concept for structures discovery from untagged corpora. *Workshop on Paradigms and Grounding in Natural Language Learning*. Adelaide, Australia. 295–299.

DELIGNE, S. AND BIMBOT, F. 1997. Inference of variable-length linguistic and acoustic units by multigrams. *Speech Comm. 23*, 223–241.

FENG, H., CHEN, K., KIT, C., AND DENG, X. 2004. Unsupervised segmentation of Chinese corpus using accessor variety. In *Proceedings of the 1st International Joint Conference on Natural Language Processing (IJCNLP)*. Sanya, Hainan. 255–261.

GAUSSIER, E. 1999. Unsupervised learning of derivational morphology from inflectional lexicons. In *Proceedings of the ACL Workshop on Unsupervised Learning in Natural Language Processing*. University of Maryland. 24–30.

GE, X., PRATT, W., AND SMYTH, P. 1999. Discovering Chinese words from unsegmented text. In *Proceedings of SIGIR*. 271–272.

GOLDSMITH, J. 2001. Unsupervised learning of the morphology of a natural language. *Computat. Linguis. 27*, 2, 153–198.

GOLDSMITH, J. 2005. An algorithm for the unsupervised learning of morphology. Tech. rep. TR-2005-06, Department of Computer Science, University of Chicago. http://humfs1.uchicago.edu/~jagoldsm/Papers/Algorithm.pdf.

GOLDSMITH, J. AND HU, Y. 2004. From signatures to finite state automata. *Midwest Computational Linguistics Colloquium*. Bloomington IN.

HAFER, M. A. AND WEISS, S. F. 1974. Word segmentation by letter successor varieties. *Inform. Storage Retriev. 10*, 371–385.

HAKULINEN, L. 1979. *Suomen kielen rakenne ja kehitys (The Structure and Development of the Finnish Language)* 4th Ed. Kustannus-Oy Otava.

HARRIS, Z. S. 1955. From phoneme to morpheme. *Language 31*, 2, 190–222. (Reprinted 1970 in Papers in Structural and Transformational Linguistics, Reidel Publishing Company, Dordrecht, Holland.)

HARRIS, Z. S. 1967. Morpheme boundaries within words: Report on a computer test. *Transformations and Discourse Analysis Papers 73*. (Reprinted 1970 in Papers in Structural and Transformational Linguistics, Reidel Publishing Company, Dordrecht, Holland.)

HU, Y., MATVEEVA, I., GOLDSMITH, J., AND SPRAGUE, C. 2005b. The SED heuristic for morpheme discovery: a look at Swahili. In *Proceedings of the 2nd Workshop of Psychocomputational Models of Human Language Acquisition*. Ann Arbor, MI. 28–35.

HU, Y., MATVEEVA, I., GOLDSMITH, J., AND SPRAGUE, C. 2005a. Using morphology and syntax together in unsupervised learning. In *Proceedings of the 2nd Workshop of Psychocomputational Models of Human Language Acquisition*. Ann Arbor, MI. 20–27.

JACQUEMIN, C. 1997. Guessing morphology from terms and corpora. In *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'97)*. Philadelphia, PA. 156–165.

JÄRVIKIVI, J. AND NIEMI, J. 2002. Form-based representation in the mental lexicon: Priming (with) bound stem allomorphs in Finnish. *Brain Lang. 81*, 412–423.

JOHNSON, H. AND MARTIN, J. 2003. Unsupervised learning of morphology for English and Inuktitut. *Human Language Technology and North American Chapter of the Association for Computational Linguistics Conference (HLT-NAACL'03)*. Edmonton, Canada.

KAZAKOV, D. 1997. Unsupervised learning of naïve morphology with genetic algorithms. *Workshop Notes of the ECML/MLnet Workshop on Empirical Learning of Natural Language Processing Tasks*. Prague, Czech Republic, 105–112.

KIT, C. 2003. How does lexical acquisition begin? A cognitive perspective. *Cognit. Science 1*, 1, 1–50.

KIT, C., PAN, H., AND CHEN, H. 2002. Learning case-based knowledge for disambiguating Chinese word segmentation: A preliminary study. In *Proceedings of the COLING Workshop SIGHAN-1*. Taipei, Taiwan. 33–39.

KIT, C. AND WILKS, Y. 1999. Unsupervised learning of word boundary with description length gain. In *Proceedings of the CoNLL99 ACL Workshop*. Bergen, Norway.

KNEISSLER, J. AND KLAKOW, D. 2001. Speech recognition for huge vocabularies by using optimized sub-word units. In *Proceedings of the 7th European Conference on Speech Communication and Technology (Eurospeech)*. Aalborg, Denmark. 69–72.

KONTOROVICH, L., RON, D., AND SINGER, Y. 2003. A Markov model for the acquisition of morphological structure. Tech. rep. CMU-CS-03-147, School of Computer Science, Carnegie Mellon University.

KOSKENNIEMI, K. 1983. Two-level morphology: A general computational model for word-form recognition and production. Ph.D. thesis, University of Helsinki.

MATTHEWS, P. H. 1991. *Morphology* 2nd Ed. Cambridge Textbooks in Linguistics.

MCKINNON, R., ALLEN, M., AND OSTERHOUT, L. 2003. Morphological decomposition involving non-productive morphemes: ERP evidence. *Cognit. Neurosci. Neuropsychol. 14*, 6, 883–886.

NAGATA, M. 1997. A self-organizing Japanese word segmenter using heuristic word identification and re-estimation. In *Proceedings of the 5th Workshop on Very Large Corpora*. 203–215.

NEUVEL, S. AND FULOP, S. A. 2002. Unsupervised learning of morphology without morphemes. In *Proceedings of the Workshop on Morphological & Phonological Learning of ACL*. 31–40.

PENG, F. AND SCHUURMANS, D. 2001. Self-supervised Chinese word segmentation. In *Proceedings of the 4th International Conference on Intelligent Data Analysis (IDA)*. Springer, 238–247.

Quirk, R., Greenbaum, S., Leech, G., and Svartvik, J. 1985. *A Comprehensive Grammar of the English Language*. Longman, Essex.

Rissanen, J. 1989. *Stochastic Complexity in Statistical Inquiry*. Vol. 15. World Scientific Series in Computer Science, Singapore.

Saffran, J. R., Newport, E. L., and Aslin, R. N. 1996. Word segmentation: The role of distributional cues. *J. Memory Lang. 35*, 606–621.

Schone, P. and Jurafsky, D. 2000. Knowledge-free induction of morphology using latent semantic analysis. In *Proceedings of the CoNLL-2000 and LLL-2000*. 67–72.

Schone, P. and Jurafsky, D. 2001. Knowledge-free induction of inflectional morphologies. In *Proceedings of the North American Chapter of the Association for Computational Linguistic Conference*.

Snover, M. G. and Brent, M. R. 2001. A Bayesian model for morpheme and paradigm identification. In *Proceedings of the 39th Annual Meeting of the ACL*. 482–490.

Snover, M. G., Jarosz, G. E., and Brent, M. R. 2002. Unsupervised learning of morphology using a novel directed search algorithm: Taking the first step. In *Proceedings of the Workshop of Morphological & Phonological Learning of ACL*. 11–20.

Wicentowski, R. 2004. Multilingual noise-robust supervised morphological analysis using the WordFrame model. In *Proceedings of the 7th ACL Special Interest Group in Computational Phonology (SIGPHON)*. Barcelona, Spain. 70–77.

Yarowsky, D., Ngai, G., and Wicentowski, R. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the 1st International Conference on Human Language Technology Research (HLT '01)*. 161–168.

Yarowsky, D. and Wicentowski, R. 2000. Minimally supervised morphological analysis by multimodal alignment. In *Proceedings of the Association for Computational Linguistics* (ACL '00). 207–216.

Yu, H. 2000. Unsupervised word induction using MDL criterion. In *Proceedings of the International Symposium of Chinese Spoken Language Processing (ISCSL)*. Beijing, China.

Received February 2005; revised July 2006; accepted October 2006 by James Martin