Empirical Comparison of Evaluation Methods for Unsupervised Learning of Morphology

Sami Virpioja[*] — Ville T. Turunen[*] — Sebastian Spiegler[**] — Oskar Kohonen[*] — Mikko Kurimo[*]

[*] *Department of Information and Computer Science, Aalto University*
P.O. Box 15400, FI-00076 Aalto, Finland. E-mail: sami.virpioja@aalto.fi
[**] *Department of Computer Science, University of Bristol*
Woodland Road, Bristol, BS8 1UB, UK. E-mail: spiegler@cs.bris.ac.uk

ABSTRACT. *Unsupervised and semi-supervised learning of morphology provide practical solutions for processing morphologically rich languages with less human labor than the traditional rule-based analyzers. Direct evaluation of the learning methods using linguistic reference analyses is important for their development, as evaluation through the final applications is often time consuming. However, even linguistic evaluation is not straightforward for full morphological analysis, because the morpheme labels generated by the learning method can be arbitrary. We review the previous evaluation methods for the learning tasks and propose new variations. In order to compare the methods, we perform an extensive meta-evaluation using the large collection of results from the Morpho Challenge competitions.*

RÉSUMÉ. *L'apprentissage non supervisé et semi-supervisé de la morphologie fournit des solutions pratiques pour le traitement des langues morphologiquement riches et requiert une intervention humaine réduite comparée aux analyseurs traditionnels basés sur des règles. L'évaluation directe des méthodes d'apprentissage utilisant des analyses de référence linguistique est importante pour leur développement, puisque l'évaluation par les applications finales prend généralement beaucoup de temps. Cependant, même l'évaluation linguistique n'est pas simple pour l'analyse morphologique complète, car les identifiants de morphèmes générés par la méthode d'apprentissage peuvent se révéler arbitraires. Nous passons en revue les méthodes d'évaluation existantes pour les tâches d'apprentissage et proposons de nouvelles variations. Afin de comparer les méthodes, nous effectuons une vaste méta-évaluation à l'aide de l'importante base de résultats provenant des compétitions Morpho Challenge.*

KEYWORDS: *Morphology, evaluation, unsupervised learning.*

MOTS-CLÉS : *Morphologie, évaluation, apprentissage non-supervisé.*

1. Introduction

A common task in natural language processing (NLP) applications such as speech recognition, machine translation and information retrieval is to construct a vocabulary and a statistical language model for all words that will be used. For many languages, particularly the morphologically rich ones, the vast amount of various inflected forms in which the words appear poses an important challenge. By using a rule-based morphological analyzer that exists already for quite many languages, most word forms can be returned to their base forms. However, these analyzers do not cover the whole language and leave out many word forms that are either rare, foreign, or ungrammatical. While the frequency of the unanalyzed types may be small in running text or speech, they might still be meaningful for the application. A special challenge is posed by less resourced languages for which the available morphological analyzers are particularly poor, as well as dialects and colloquial and historical languages.

A large variety of algorithms for unsupervised morpheme analysis have been presented during the last ten years.[1] Due to the amount of work required by evaluation in applications, the algorithms are most often evaluated directly based on a linguistic reference analysis. Various automatic evaluations have been proposed, depending on the task of the algorithm and the available reference analyses. For morphological segmentation, the simplest solution is to calculate how well the segmentation boundaries correspond to the ones in the reference analysis (e.g., Creutz and Lindén, 2004; Kurimo *et al.*, 2006). For finding morphologically related words, the evaluations are usually based on pairs or groups of words that share the same stem or root in the reference (e.g., Schone and Jurafsky, 2000; Snover *et al.*, 2002).

The most general task, full morphological analysis of word forms, is hard not only for the algorithms, but also for the evaluation. If the learning task is unsupervised or semi-supervised, it cannot be expected that the algorithm comes up with morpheme labels that exactly correspond to the ones designed by linguists. For example, the words “foot” and “feet” might both contain the morpheme “foot_N” in an English reference analysis. Also the applied algorithm should discover a morpheme that occurs in both these word forms, but it may be labeled as “FOOT”, “morpheme784”, “foot”, or something else (Kurimo *et al.*, 2008). The problem is similar to the one in the evaluation of unsupervised part-of-speech (POS) tagging (see, e.g., Christodoulopoulos *et al.*, 2010). However, there the number of labels per word is exactly one.

Spiegler and Monson (2010) have listed computational and linguistic criteria for an automatic evaluation method dealing with full morphological analyses. The method should be quicker and easier to compute than large NLP tasks (*readily computable*), reflect accurately the distribution of predicted and true morphemes and be difficult to game (*robust*), the results should *correlate* to the performance in NLP tasks, and be useful for identifying the strengths and weaknesses of the algorithm (*readily interpretable*). Moreover, it should account for *morphophonology*, *allomorphy*, *syn-*

[1] For a recent survey, see Hammarström and Borin (2011).

cretism, and *ambiguity*. So far, only the metric used in the recent Morpho Challenge competitions (Kurimo *et al.*, 2008; Kurimo *et al.*, 2009; Kurimo *et al.*, 2010c; Kurimo *et al.*, 2010b) and EMMA by Spiegler and Monson (2010) have met all or most of these criteria.

The series of the Morpho Challenge competitions, started in 2005, has supported the research for unsupervised morpheme analysis by providing annual evaluations for shared tasks using shared training data for various languages (for an overview, see Kurimo *et al.*, 2010a). The goal has been to develop unsupervised and language independent machine learning algorithms that could discover morphemes from large amount of given raw text data. From 2007 onwards, the tasks have been designed for algorithms performing full morphological analyses. In 2010, small amounts of labeled data were provided to support semi-supervised algorithms, relevant for this task particularly because small samples of morphologically labeled words are often easy to obtain. So far, more than fifty algorithms have been evaluated and compared using the shared tasks. They have been evaluated not only by comparing the obtained morphemes to the linguistic ones, but also by testing them in real NLP applications using state-of-the-art technology.

In this article, we utilize the large database of results from the Morpho Challenges to perform the most extensive meta-evaluation of unsupervised learning of morphology so far. While we review the previous evaluation methods also for morphological segmentation and clustering, the main focus of this article are the evaluation methods for unsupervised morphological analysis.

The structure of the rest of the article is as follows: in section 2, we present the central terminology and issues in morphology and machine learning. In section 3, we review the evaluation methods proposed for the unsupervised learning of morphology, including the Morpho Challenge evaluation and EMMA, and propose a few new variants for the evaluation of full morphological analyses. In section 4, we describe the setup and the results of the experimental meta-evaluation. We consider correlations to information retrieval and machine translation applications, as well as robustness, interpretability, stability, and computational complexity of the methods. In section 5, we conclude the work and give a few recommendations for those who need to evaluate their algorithms.

2. Background on Morphology and Machine Learning

2.1. *Morphology*

Morphology is the study of internal structure of words.[2] A common way to look at the structure is to observe *morphemes*, the smallest meaning-bearing units of the

[2] For a text-book description on morphology, see, e.g., Matthews (1991) or Chapter 3 in Jurafsky and Martin (2008).

language, and how they are combined to form words. To formalize the different problems related to learning of morphology, we use the idea of two-level morphology by Koskenniemi (1983). The two levels of representation are lexical level, which has the abstract morphemes (each reflecting a certain meaning), and surface level, which has the phonemes or letters that are the realizations of the morpheme. The surface realizations of morphemes are called *morphs*. For example, *walked* has two morphs, *walk* and *ed*, corresponding to abstract morphemes that refer to the meaning of walking and past tense, respectively. Sometimes a morpheme that does not have a surface realization (e.g., one representing the singular form of an English noun) may be added to the lexical representation; these are called *null morphemes*.

Morphemes are divided into *free* or *bound* morphemes depending on whether they can occur independently as a word form or not. Bound morphemes are usually *affixes* that are attached to *stems*. Stem is the word without any inflections, but it may be a compound word or include derivative morphemes, whereas *root* is the minimal part of the stem that cannot be reproduced. For example, *buildings* has the stem *building* and the root *build*. Affixes are further divided into *prefixes*, *suffixes*, *infixes*, and *circumfixes*, depending on whether they occur before, after, in between, or both after and before of the stem, respectively.

There are several ways how the morphemes of the word are realized in the surface form. The simplest one is *agglutination*, where the morphs are concatenated together, as in *walk+ed*. This contrasts to *fusion*, in which the surface form does not have separable morphs corresponding to the morphemes. For example, *run* and past tense are realized as a single morph *ran*. Another type of non-concatenative process is *transfixation*, present for example in Arabic, where consonantal roots are modified by vowel patterns.

Even in concatenative case, the interaction between morphological and phonetic processes (*morphophonology*) may produce different morphs for the same morpheme. For example, the past tense of *invite* has morph *d* instead of *ed*, and and plural of *wife* is *wi***v***e+s*, not *wi***f***e+s*. Two morphs of the same morpheme are called *allomorphs*. In addition to phonological processes that occur at morph or word boundaries (called *sandhi*), allomorphic variations are produced by such phenomena as consonant gradation (Finnish example: *ta***kk***a*–*ta***k***a+n*), vowel harmony (Finnish example: *alu+***ssa**–*äly+***ssä**), and wovel variations called *ablaut* (English example: *s***i***ng*–*s***a***ng*–*s***u***ng*).

The case in which several morphemes have the same surface realization is called *syncretism*. A common example is English plural and 3rd person singular, which are both realized by the suffix *s* (or *es*). Syncretism is one cause for morphological *ambiguity* of the word form. For example, *bites* may either be noun in plural form or verb in 3rd person singular form. However, the full word form can be ambiguous even without syncretism. For example, Finnish word *istuin* may refer either to the noun "seat" or the verb "sit" in 1st person past tense ("I sat").

The richness of morphological phenomena varies between languages. Specifically, some languages are *analytic* or *isolating*, with one-to-one correspondence between words and morphemes, and some are *synthetic*, with often many morphemes per word. Synthetic languages can furthermore be divided into *agglutinative* (or *concatenative*) and *fusional* languages, depending on whether the morphs of a word are clearly distinguishable from each other or not. Naturally, the characteristics of many languages are in between these categories. Of the languages used in the experiments of this article, English is moderately analytic and often considered as a fusional language. Finnish and Turkish are highly synthetic and agglutinative languages. Also German is mainly agglutinative, and more synthetic than English, but less synthetic than Finnish or Turkish.

2.2. *Supervised and Unsupervised Learning*

Machine learning algorithms can be divided into *supervised* and *unsupervised* learning depending on what kind of training data they use (Alpaydin, 2004). In the tasks of morphology learning, *input data* consists of word forms. Depending on the task, the *output data* may be stems or lemmas of the words, segments of the words, or labels for the morphemes of the words. Regardless of the output type, data with samples of input-output pairs is called *labeled* (or annotated) data, and data with only input samples is called *unlabeled* (or unannotated) data.

Supervised algorithms try to learn a mapping from the input variable X to output variable Y given pairs of data samples (x_i, y_i), i.e., labeled data. For the task of morphology learning, the usefulness of this kind of algorithms is limited: in order to have enough training data samples, there already needs to be a morphological analyzer for the language. Of course, a sophisticated method might be able to generalize the analyses to samples not correctly identified by the original analyzer.

In unsupervised learning, the samples of the desired output Y are not available. The algorithm needs to assume some statistical regularities in the input data, and use those to create a model for the input that can be used also for predicting the output. In morphology learning, large amounts of unlabeled data (i.e., words) are easy to get. The problem is how to use the information in the word forms to get the desired output.

Semi-supervised algorithms deal with settings where both unlabeled and labeled data are available (for a survey, see Zhu, 2005). Usually the amount of labeled data is remarkably smaller than the amount of unlabeled data, so that supervised algorithms do not have enough data to get good results. This setting is very relevant for learning of morphology, as small amounts of labeled data are easy to get by manual annotation.

In the case of morphology learning, there are also many settings where all the training data is *partially* labeled. That is, the label y_i in the data samples is not directly the desired output of the algorithm, but some part of it. The partial information can be, e.g., suffixes (Yarowsky and Wicentowski, 2000), stems (Shalonova *et al.*, 2009), or even all morphs of the word, if the desired output is morphemes. See Spiegler (2011) for a more extensive categorization of different types of scenarios.

2.3. *Tasks in Learning Morphology*

There are three common tasks for morphological processing in NLP applications: segmenting words into morphs, identification of morphologically related word forms, and performing full morphological analysis, that is, finding the morphemes of the words.

2.3.1. *Morphological Segmentation*

Morphological segmentation (or word decomposition) is a useful approach for specific applications and languages. First, in applications such as speech recognition, it may be enough to deal with the surface forms of the words. Second, if the language is mainly agglutinative, the set of morphs is not essentially different from the set of morphemes, and thus the result of segmentation is close to the full analysis.

Direct evaluation of a morphological segmentation is straightforward given that reference segmentations are available. The output for a segmentation algorithm is an ordered list of substrings, morphs, and the strings should be the same that are found from the reference segmentation.

2.3.2. *Clustering of Related Word Forms*

From the machine learning viewpoint, finding morphologically-related words can be considered as *clustering*: a set of words are grouped together either by their stem or root. The task is important especially for improving the recall of an information retrieval system: if the user searches for “election”, it is likely that also the documents containing, e.g., “elections” or “elect” are relevant. For many languages, the standard way of finding related word forms is suffix stripping: if the morphology is relatively simple, a set of hand-crafted rules are often enough to get reasonably good results.

When clustering morphologically related word forms, the output is one label per each word. While the labels of the proposed result and the reference result have to be matched in order to do evaluation, this is still a relatively simple setting: either two words are in the same cluster or in different clusters.

2.3.3. *Morphological Analysis*

The most challenging task is the full morphological analysis. In order to find the correct morphemes, morphological analyzers need to deal with complex phenomena such as allomorphy and syncretism. In consequence, the analyzers have traditionally been rule-based and designed by linguists.

For morphological analysis, the output is ordered list of labels, and in the case of unsupervised learning, there is no direct evidence on which predicted label is related to which reference label. Even in the (semi-)supervised case, not all labels are found in the labeled training data. The problem in evaluation can be avoided only if there is a systematic way to label the morphemes to be equivalent to those in the reference analysis.

To illustrate the problem, Table 1 shows how various unsupervised and semi-supervised methods (described later in section 4.4) analyze the words *reproduces* and *vulnerabilities*. Of the present methods, only Morfessor Baseline, Morfessor S+W and Promodes 2010 are strictly based on segmentation. Some others, such as Bernhard 2 2007 and Morfessor CatMAP, do mainly segmentation, but also try to differentiate between stem and affix morphemes. Evidently, there is no way to evaluate the analyses of individual words without observing the other related words. For example, that *reproduced* has a morpheme *produc_B* is a useful piece of information only if the forms *produce*, *produced*, *producing*, etc., have the same morpheme.

Table 1. *Examples of the analyses of different algorithms for English words "reproduces" and "vulnerabilities".*

Method	*Analysis for "reproduces"*	*Analysis for "vulnerabilities"*
Allomorfessor 2009	re produce s	vulnerability ies
Bernhard 2 2007	re_P produc_B e_S s_S	vulnerabilit_B i_L e_S s_S
Bordag 5a 2007	re pro.prop duc es	vulnerabilities
DEAP MDL-CAT	re_p produc_a es_pl	vulnerabil_n iti_s es_pl
Lignos Base Inference	REPRODUCE +(s)	VULNERABILITY +(ies)
MAGIP 2010	REPRODUC ES	VULNERABI L ITI ES
Morfessor Baseline	re produce s	vulner + abilities
Morfessor CatMAP	re/PRE + produce/STM + s/SUF	vulner/STM + abilities/STM
Morfessor S+W	re produce s	vulner abilities
Morfessor S+W+L	re_p produce_N +PL	vulner_N abilities_N
MorphAcq 2010	+#re -produc- +e# +s#	-vulnerab- +ilities#
MorphoNet 2009	produce _s re_s re_	vulnerabilty _ies
ParaMor Mimic	reproduc +e +s	vulner +a +bilit +ie +s
Promodes 2010	re pro du c e s	vul nera b i l iti e s
RALI-COF	prod uce re s	vulnerability ies
Reference	re_p produce_V +3SG	vulnerable_A ity_s +PL

3. Evaluation Methods

There are two main approaches to evaluate the methods for learning morphology. In *direct* evaluation, we directly study the produced morphological analyses, either manually or by comparing to external data. In *indirect* evaluation, we evaluate the effect on using the method as a part of a larger system, typically an application such as information retrieval or speech recognition. While the real usefulness of the method is measured by how well it can help the applications of natural language processing, the application evaluations are often too time consuming to use during the development of the method. Apart from being simpler, direct evaluations often provide more information on the specific problems of the evaluated method.

3.1. *Direct Evaluations*

There are two basic ways to do direct evaluation. Either the results are manually evaluated by experts of the language, or the results are compared to a linguistic reference ("gold standard") analysis using an automatic evaluation method.

Regardless of the details of the evaluation method, the measures of evaluation are often the same. Similarly to IR tasks, there are two sets to consider: *predicted* items P (analogous to retrieved documents) and *reference* items R (analogous to relevant documents). The intersection of the sets, $P \cap R$, tells which of the predicted items were correct. If the size of the intersection is normalized with the number of predicted items, we get the *precision* (Pre), and if it is normalized with the number of reference items, we get the *recall* (Rec):

$$\text{Pre} = \frac{|P \cap R|}{|P|}; \quad \text{Rec} = \frac{|P \cap R|}{|R|}, \qquad [1]$$

where $|A|$ denotes the size of the set A. Precision measures how many of the predicted items are correct, whereas recall measures how many of the reference items are predicted. To get a global estimate, the average is taken over a set of words, sometimes weighted by the number of items $|P|$ and $|R|$.

If the task is morphological clustering, P and R contain only the proposed root or lemma of the word. As $|P| = |R| = 1$, precision and recall are equal and, for a single word, either one or zero. The global measure is accuracy, the proportion of correct answers to all answers.

If the task is morphological segmentation, P and R are either segmentation points (section 3.1.2 below) or sets of morphs. In morphological analysis, P and R are usually sets of morphemes. In all of these cases, both precision and recall have to be observed. Otherwise, the evaluation can be *gamed* by either predicting as few items as possible, which gives high precision but low recall, or predicting as many items as possible, which gives high recall but low precision. In the case of segmentation algorithms, the former is referred to as *undersegmentation* and the latter as *oversegmentation*.

To get a single measure that includes the aspects of both precision and recall, they are combined using harmonic mean, resulting in *F-score* or *F-measure*:

$$\text{F} = \frac{2}{\frac{1}{\text{Pre}} + \frac{1}{\text{Rec}}} = \frac{2 \times \text{Pre} \times \text{Rec}}{\text{Pre} + \text{Rec}} \qquad [2]$$

In addition to the balanced F-score, we consider also the more general F$_\beta$-score:

$$\text{F}_\beta = \frac{(1 + \beta^2) \times \text{Pre} \times \text{Rec}}{\beta^2 \times \text{Pre} + \text{Rec}}, \qquad [3]$$

where $\beta > 1$ gives more weight to recall and $\beta < 1$ gives more weight to precision.

3.1.1. *Manual Evaluation*

While manual inspection is very useful for developing a method, the amount of the work involved restricts its usage. Moreover, the decisions on correct and incorrect answers can be subjective, as exemplified by Goldsmith (2001). Usually a binary true/false categorization is too coarse, and some intermediate categories are added. For example, Goldsmith (2001) uses four categories ("good", "wrong analysis", "failed to analyze", and "spurious analysis") and Creutz and Lagus (2002) three categories ("correct", "incomplete", and "incorrect").

3.1.2. *Segmentation Boundaries*

In many languages, the main parts of the morphological processes are concatenative. For such languages, the problem can be reduced to finding surface forms of the morphemes, morphs, that the words contain. Equivalently, the task is to predict whether there is a *morph boundary* between each of the two successive letters of a word or not. Given a reference segmentation, precision (how many of the predicted segmentation points were correct) and recall (how many of the correct segmentation points were found) can be calculated. This kind of evaluation has been done already by Hafer and Weiss (1974), who did several segmentation methods based on the idea of letter successor variety (LSV) by Harris (1955). More recently, evaluation based on segmentation boundaries has been used in Morpho Challenge 2005 competition (Kurimo *et al.*, 2006), and by many separate studies such as Creutz and Lagus (2007), Dasgupta and Ng (2007), Snyder and Barzilay (2008) and Poon *et al.* (2009). In addition to neglecting non-concatenative processes such as allomorphy, there is an amount of subjectivity involved at judging the correct segmentation point.

3.1.3. *Co-occurrence Analysis*

If non-concatenative processes are taken into account, and the task is fully unsupervised, the predicted morphemes can be arbitrary. This prevents calculation of precision and recall directly using the intersection of the sets of predicted and reference morphemes as in Equation 1.

Disregarding the ordering of the morphemes inside each word, the analyses for a set of words can be represented as a *bipartite graph* $G = (M, W; E)$ (Spiegler and Monson, 2010). The graph has two disjoint sets of vertices, morphemes $M = \{m_1, \ldots, m_n\}$ and words $W = \{w_1, \ldots, w_m\}$, and edges $e(m_i, w_j)$ that connect vertices in M to vertices in W. Such a graph is illustrated by Figure 1. The edges can have weights corresponding to how many times the morpheme m_i occurs in the word w_j.[3] Equivalently to the graph, the analyses can be presented as a morpheme-word co-occurrence matrix $\mathbf{A}$, where the element a_{ij} is the weight of the corresponding edge or zero if the edge is missing.

[3] In some languages, the same morpheme can occur more than once in a word due to reduplication or compounding. For example, Finnish word *maankuoren* ("of earth's crust") contains two genitives marked by suffix *n*.

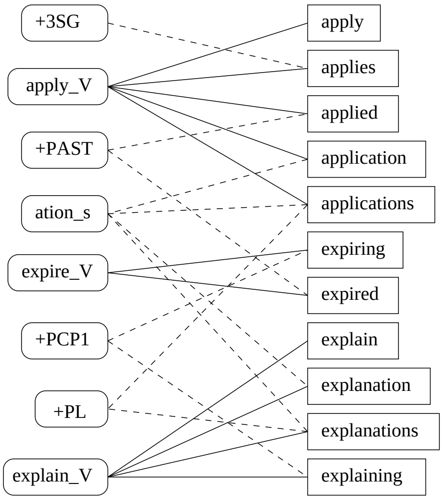

Figure 1. *Bipartite morpheme-word graph for a set of morphological gold standard analyses of English words. An edge between a morpheme and a word indicates that the word contains the morpheme. Edges to suffixes are drawn with dashed lines.*

Considering predicted and reference analyses for the same set of words, we have two bipartite graphs that have the same word vertices but different morpheme vertices. Evidently, the evaluation must be based on the information of which words are linked to the same morphemes. Furthermore, the methods relate to solving graph *isomorphism*: two sets of analyses are equivalent only if the corresponding graphs are isomorphic, i.e., there exist a bijection between the vertices of the two graphs.

In this section, we consider methods that do not explicitly match the predicted and reference morphemes, but study co-occurrences of morphemes in the words. While Spiegler and Monson (2010) called this type of approach *soft isomorphic analysis*, we use the term *co-occurrence analysis* to distinguish it from methods that use soft matching for the predicted and reference morphemes (discussed in section 3.1.4).

The bipartite morpheme-word graph can be transformed into a *word graph* by removing the morpheme vertices and replacing each pair of edges $e(m_i, w_j), e(m_i, w_k)$ by edge $e(w_j, w_k)$. Figure 2 illustrates the word graph corresponding to the bipartite graph in Figure 1. An equivalent word matrix is obtained by the product $\mathbf{A}^{\mathrm{T}}\mathbf{A}$ of the morpheme-word matrix $\mathbf{A}$. As the set of the vertices are now the same regardless of the evaluated method, it is enough to compare the edges between the vertices.

Given two word graphs, one from reference analyses and one from predicted analyses, we can compare the sets of words that are connected. Specifically, recall can be determined from the number of edges that are in the reference graph but are not in the predicted graph, and precision from the number of edges that are in the predicted

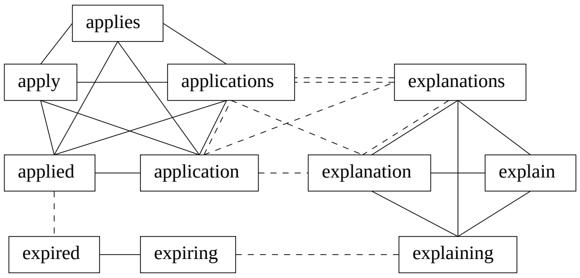

Figure 2. *Word graph for a set of morphological analyses of English words. An egde between two words indicates a co-occurring morpheme. Edges corresponding to suffixes are drawn with dashed lines.*

graph but are not in the reference graph. Intuitively, low recall in co-occurrence based metrics then indicates that you are missing co-occurrences, and low precision indicates that you have spurious co-occurrences. As an example, consider the mistakes that a segmentation algorithm might make for the example in Figure 2. On one hand, oversegmentation, such as predicting *ex-* to be a morpheme in all the word forms that start with it, will add many new edges to the graph and thus decrease precision. Recall is either unchanged or increased if some correct edges are added by chance. Similarly, not distinguishing between the two different morphemes (plural and 3$^{\text{rd}}$ person singular) for suffix morph *-s* will add incorrect links between *applies—applications* and *applies—explanations*. On the other hand, leaving a true suffix together with a stem, e.g., not segmenting *-ing* from *expiring* and *explaining*, will remove true edges and thus decrease recall. Correct edges will also be missed if allomorphs such as *apply*, *appli*, and *applic* are left as separate morphemes.

For algorithms that try to find morphologically related words (i.e., those having the same stem or root), the evaluation is relatively straightforward. Schone and Jurafsky (2000; 2001) study *conflation sets* of the words, that is, the sets of words which share the same stem. For example, removing all the affix edges from the graph in Figure 2 results in three conflation sets: *{apply, applied, applies, application, applications}*, *{explain, explaining, explanation, explanations}*, and *{expired, expiring}*. For each word, Schone and Jurafsky sum the number of correct (C), inserted (I), and deleted (D) words compared to the reference conflation set. The numbers are summed over the words, and precision ($C/(C+I)$) and recall ($C/(C+D)$) are calculated. Snover *et al.* (2002) use a similar setting, but instead of observing the groups of words, they go over the *stem relations*, i.e., pairs of words that share the same stem. Precision gives how many of the predicted relations were correct, and recall how many of the relations found from the reference analysis were found. Also Baroni *et al.* (2002) discover morphologically related pairs. As the result of the algorithm is a ranked list of pairs, they evaluate it by calculating the precision (amount of correct pairings with respect to the reference analysis) over different numbers of pairs.

There is less work on evaluation metrics for full morphological analyses, where both stems and affixes should match to the reference. The most notable is the method developed by Mathias Creutz that has been used in the Morpho Challenges from 2007 onwards (Kurimo *et al.*, 2008). The method was slightly revised for Morpho Challenge 2009 (see Kurimo *et al.*, 2010c). We will refer to it as the MC evaluation.

The first step in the MC evaluation is to randomly sample a number of *focus words* from the set of words for which both predicted and reference analyses are available. Then, for each predicted morpheme of each focus word, another word that has the same morpheme is sampled. Morphemes that do not occur in any other words are excluded. The result is a set of word pairs that have at least one morpheme in common. The precision for one focus word is the proportion of its word pairs that have a common morpheme also according to the reference analyses. However, if a word pair shares multiple morphemes, the reference analysis has to have an equal amount of shared morphemes in order to get full points. Otherwise, a corresponding fraction of points is given. The overall precision is the average over the focus words. Similarly, focus words and their pairs are sampled from the reference, and the recall is the average proportion of word pairs that have a common morpheme also in the predicted analyses.

The rationale behind the two-phase sampling of word pairs is that because two random words rarely share a common morpheme, the evaluation should concentrate on the pairs that do. Considering the word graph representation (Figure 2), the idea is simply to sample edges of the graph for evaluation. While this approach is well-motivated and efficient for large graphs that cannot be compared as a whole, it has two drawbacks. First, as the word pairs that are sampled for calculating precision are dependent on the predicted analyses, two different algorithms will have different word pairs in the evaluation. Second, the approach is inconvenient if the reference set has only a small number of words, as it does not use all the information in the known analyses.

The MC evaluation also allows alternative analyses for the possibly ambiguous word forms. If a word in the predicted analyses has several alternatives, the precision for the word is the average over them. If the reference has several alternative analyses, the one that gives the highest precision is selected. The same holds the other way round for recall, as precision and recall are calculated symmetrically. However, this allows a way of improving the recall artificially by adding alternative analyses for the predictions: for each word, precision is the average precision over the alternative analyses, but recall is the best one (Kurimo *et al.*, 2010a).

The limitations of the MC evaluation have been analyzed in more detail by Spiegler (2011). To improve on especially the two problems mentioned above, we propose a new set of evaluation methods based on co-occurrence analysis, referred to as CoMMA[4].

[4] CoMMA stands for Co-occurrence based Metric for Morphological Analysis.

Let us first assume that we have only one analysis per word. Let V be the set of words for which we have the analyses, P_i the set of predicted morpheme labels, R_i the set of reference morpheme labels for the i:th word, and $\mathbf{P}$ and $\mathbf{R}$ matrices of size $|V| \times |V|$ where

$$p_{ij} = |P_i \cap P_j|, \qquad r_{ij} = |R_i \cap R_j|. \qquad [4]$$

That is, p_{ij} is the number of predicted labels and r_{ij} is the number of reference labels that are both in word i and j. Clearly, if the analyses are isomorphic, $\mathbf{P}$ and $\mathbf{R}$ will be equal. A simple measure for the error would be the 1-norm distance $|\mathbf{P} - \mathbf{R}| = \sum_i \sum_j |p_{ij} - r_{ij}|$. However, we can also derive precision and recall measures similar to those of the MC evaluation. For each word, let the number of words with at least one common morpheme with word i be $n_i = |\{j : p_{ij} > 0\}|$ and $m_i = |\{j : r_{ij} > 0\}|$, and the number of words that have at least one common morpheme with *any* word $v_p = |\{i : n_i > 0\}|$ and $v_r = |\{i : m_i > 0\}|$. The overall precision and recall are

$$\text{Pre} = \frac{1}{v_p} \sum_{i:n_i>0} \frac{1}{n_i} \sum_{j:p_{ij}>0} \frac{\min(p_{ij}, r_{ij})}{p_{ij}}; \qquad [5]$$

$$\text{Rec} = \frac{1}{v_r} \sum_{i:m_i>0} \frac{1}{m_i} \sum_{j:r_{ij}>0} \frac{\min(r_{ij}, p_{ij})}{r_{ij}}. \qquad [6]$$

For example, if there are two morphemes that are shared between words i and j in the predicted analyses ($p_{ij} = 2$) and one morpheme in the reference analyses ($r_{ij} = 1$), the precision increases 0.5 point and the recall increases one point ($\min$ ensuring that the maximal points are one). One option is to set the diagonals of the matrices $\mathbf{P}$ and $\mathbf{R}$ to zeros, that is, $p_{ii} = r_{ii} = 0$ for all i. This excludes the isolated words that do not have a common morpheme with any other words from the evaluation.

Next, let us consider the case of several alternative analyses. If P_{ik} is the k:th alternative for the i:th word in the predicted analyses, and R_{il} similarly for the reference analyses, the simplest way to proceed is to reduce the alternatives by taking the maximal co-occurrence counts:

$$p_{ij} = \max_k \max_l |P_{ik} \cap P_{jl}|, \qquad r_{ij} = \max_k \max_l |R_{ik} \cap R_{jl}| \qquad [7]$$

This ensures that adding more alternatives in the prediction will increase p_{ij}:s, thus generally improving recall but degrading precision. We refer to this method as "CoMMA-B".

We can also derive a measure that directly penalizes for a wrong number of alternatives. For $\mathbf{P}$ and $\mathbf{R}$ to be comparable, we cannot expand both the rows and columns to include the alternative analyses. Instead, we add them only to the rows:

$$p_{(ik)j} = \max_l |P_{ik} \cap P_{jl}|, \qquad r_{(ik)j} = \max_l |R_{ik} \cap R_{jl}|, \qquad [8]$$

where (ik) denotes the index for the k:th analysis of the i:th word. The numbers of words with shared morphemes are $n_{ik} = |\{j : p_{ikj} > 0\}|$ for the predicted analyses and $m_{ik} = |\{j : r_{ikj} > 0\}|$ for the reference analyses. Let $o_i = |\{k : n_{ik} > 0\}|$ and $q_i = |\{k : m_{ik} > 0\}|$ be the number of alternative analyses for word i in predicted and reference analyses, respectively. Now v_p and v_r are defined as $v_p = |\{i : o_i > 0\}|$ and $v_r = |\{i : q_i > 0\}|$. The overall precision and recall are

$$\text{Pre} = \frac{1}{v_p} \sum_{i:o_i>0} \frac{1}{o_i} \max_{A_i} \sum_{k:n_{ik}>0} \frac{1}{n_{ik}} \sum_{j:p_{ikj}>0} a_{ikl} \times \frac{\min(p_{ikj}, r_{ilj})}{p_{ikj}}; \qquad [9]$$

$$\text{Rec} = \frac{1}{v_r} \sum_{i:q_i>0} \frac{1}{q_i} \max_{A_i} \sum_{k:m_{ik}>0} \frac{1}{m_{ik}} \sum_{j:r_{ikj}>0} a_{ikl} \times \frac{\min(r_{ikj}, p_{ilj})}{r_{ikj}}, \qquad [10]$$

where A_i is an assignment matrix between predicted and reference alternatives of the i:th word. That is, we want $\sum_k a_{ikl} \leq 1$, $\sum_l a_{ikl} \leq 1$, and $a_{ikl} \in \{0, 1\}$ for all i, k, and l. The best assignment can be solved using the Hungarian algorithm (Kuhn, 1955; Munkres, 1957). The cost for assigning the k:th and the l:th alternative of the i:th word is set to one minus the F-score for the pair of analyses, using the precision and recall as defined above. This results in the best average F-score. The assignment is quick regardless of the $O(n^3)$ time complexity for the $n \times n$ matrix, because the number of alternatives is usually low. We refer to this version as "CoMMA-S".

3.1.4. *Morpheme Assignment*

The evaluations based on morpheme assignment try to find a one-to-one or one-to-many assignments between the predicted and reference morphemes. One-to-one matching can be considered as hard isomorphic analysis. One-to-many (or many-to-one) matching is often simpler to solve than one-to-one matching, but as they are easier to game by providing a low (or high) number of predicted morphemes, the evaluation setup becomes more complicated.

In the case of supervision, the assignment is often known. E.g., Yarowsky and Wicentowski (2000) study finding the roots of inflected word forms, including irregular inflections. Since the input data consists of a finite set of candidate roots for the algorithm, they can directly calculate the proportion of correct roots.

Creutz and Lagus (2002) and Creutz (2003) use the Viterbi algorithm to align predicted morph segmentation to a linguistic morpheme analysis. To calculate a distance between a predicted morph m and a morpheme label l, they use the measure $d(m, l) = -\log \frac{c_{m,l}}{c_m}$, where $c_{m,l}$ is the number of word tokens in which the morph m is aligned with the label l and c_m the total count of m. As one-to-many mapping from morphs to labels is accepted, a separate training and test set are needed to avoid overfitting predictions that are undersegmented (i.e., a single predicted morph is mapped to all the labels of the word). The final measure is the alignment distance in the test set. A similar cross-validation setting has been applied in the evaluations of unsupervised POS tagging (Gao and Johnson, 2008; Christodoulopoulos *et al.*, 2010).

Spiegler and Monson (2010) propose an evaluation method, EMMA, which applies a one-to-one assignment. In EMMA, each predicted morpheme is matched for each morpheme in the reference. One-to-one matching allows direct calculation of precision (how many morphemes in the proposed analysis are in the reference analysis) and recall (how many morphemes in the reference analysis are in the proposed analysis) for each word (Equation 1).

The assignment problem can be described using a bipartite graph, where one set of vertices correspond to the reference morphemes and another set of vertices to the predicted morphemes. Let R_k and P_k be the reference and predicted morphemes for word w_k, respectively. An edge $e(m_i, m_j)$ exists between reference morpheme m_i and predicted morpheme m_j if they both are in the analysis of at least one same word w_k. The weight c_{ij} of the edge is the number of such words:

$$c_{ij} = |\{k : m_i \in P_k \wedge m_j \in R_k\}|. \qquad [11]$$

Given morpheme-word graphs for the reference and predictions, such a morpheme graph can be formed by removing the word vertices and replacing pairs of edges $e(m_i, w_k)$ and $e(m_j, w_k)$ by $e(m_i, m_j)$. For example, if we have the reference analysis corresponding to Figure 1 and segmentations *app+ly*, *app+lie+s*, *applied*, *application*, *application+s*, *expir+ing*, *expir+ed*, *explain*, *expla+nation*, *expla+nation+s*, and *explain+ing* for the same words, the resulting graph is the one in Figure 3.

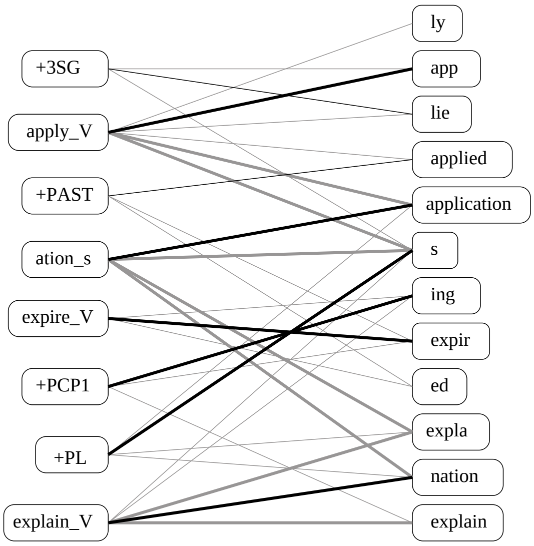

Figure 3. *Bipartite morpheme graph for reference (left) and predicted (right) morphemes of English words. An egde between two morphemes indicates that there is one (thin lines) or two (thick lines) words that have the left morpheme in reference analysis and the right morpheme in the predicted analysis. Black lines show one possible assignment that maximizes the target criterion in EMMA.*

Matching two morphemes that have an edge increases both precision and recall, the more the larger the weight of the edge is. Thus the task is to select such an assignment that maximizes the sum of the weights of the selected edges. Mathematically, it is defined as

$$\arg\max_{\mathbf{B}} \sum_{i,j} (c_{ij} \times b_{ij}) \quad \text{s.t.} \quad \sum_{i} b_{ij} \leq 1, \sum_{j} b_{ij} \leq 1, b_{ij} \in \{0,1\}. \qquad [12]$$

$\mathbf{B}$ is a binary assignment matrix, where $b_{ij} = 1$ indicates that morpheme m_i in the predicted analysis is matched to morpheme m_j in the reference analysis. The black edges in Figure 3 show one assignment that maximizes the criterion for the example graph. There are several other assignments that give the same sum: for example, *+PAST* could be matched to *ed* instead of *applied.* This is, however, less likely with larger data sets for which the weights are more varied.

To permit several alternative analyses per word in EMMA, c_{ij} is redefined as the average over all the combinations of the alternative analyses. After obtaining $\mathbf{B}$, a one-to-one mapping between the alternatives is optimized.

While EMMA is a robust measure that correlates well on the application evaluations (Spiegler and Monson, 2010; Kurimo *et al.*, 2010b), it has one major drawback. The time complexity of solving the assignment in Equation 12 is $O(n^3)$ for n morphemes using the Hungarian algorithm [5], so the computation time increases rapidly with the size of the evaluation set.

We introduce a modified version of the EMMA (referred to as "EMMA-2"), which solves this problem by replacing the single one-to-one assignment problem with two many-to-one problems. The idea is that failing to join two allomorphs (e.g., plural suffixes *-s* and *-es* in English) does not need to degrade precision. Thus, when calculating precision, we apply a many-to-one mapping, where several predicted morphemes may be assigned to one reference morpheme. Similarly, failing to distinguish between surface-identical syncretic morphemes (e.g., plural *-s* and 3rd person singular *-s* in English) does not need to degrade recall, so one-to-many mapping is applied there. A potential problem is that the relaxed mappings may facilitate gaming.

The modified assignment problems in EMMA-2 are

$$\mathbf{B}_{\text{Pre}} = \arg\max_{\mathbf{B}} \sum_{i,j} (c_{ij} \times b_{ij}) \quad \text{s.t.} \quad \sum_{j} b_{ij} \leq 1, b_{ij} \in \{0,1\}, \qquad [13]$$

for precision and

$$\mathbf{B}_{\text{Rec}} = \arg\max_{\mathbf{B}} \sum_{i,j} (c_{ij} \times b_{ij}) \quad \text{s.t.} \quad \sum_{i} b_{ij} \leq 1, b_{ij} \in \{0,1\}, \qquad [14]$$

[5] Spiegler and Monson (2010) use a general integer linear programming software. Integer linear programming is a NP-hard problem (Karp, 1972), so the time complexity cannot be any better than with the Hungarian algorithm.

for recall. In contrast to the assignment problem in EMMA, solving these problems is very simple, as the best match for each reference or predicted morpheme can be selected independently of the others. For precision, we set $b_{ij} = 1$ for certain i only if $j = \arg\max_j c_{ij}$. For recall, we set $b_{ij} = 1$ for certain j only if $i = \arg\max_i c_{ij}$. This requires $O(nm)$ time for n predicted and m reference morphemes. Other aspects of the evaluation (dealing with alternative analyses and calculating precision and recall) are done similarly to the original EMMA.

3.1.5. *Information-Theoretic Methods*

Instead of heuristic evaluations based on co-occurrence analyses or morpheme assignment, it would be desirable to have a measure that would directly tell how much information is preserved or changed when comparing the predicted analyses $\mathcal{P}$ to the reference analyses $\mathcal{R}$. Indeed, Rosenberg and Hirschberg (2007) have proposed an entropy-based V-measure, which resembles F-measure. It is the harmonic mean of homogeneity h (analogous to precision) and completeness c (analogous to recall):

$$h = \frac{I(\mathcal{P}, \mathcal{R})}{H(\mathcal{P})}; \quad c = \frac{I(\mathcal{P}, \mathcal{R})}{H(\mathcal{R})}, \qquad [15]$$

where H is entropy (analogous to the size of the set) and I mutual information (analogous to the size of the intersection of the sets). V-measure and other information-theoretic measures have been applied to unsupervised POS tagging. Christodoulopoulos *et al.* (2010) compared several evaluation measures for this task, and found V-measure to be the most stable one.

Why not use information-theoretic measures for evaluating morphological analyses? Note that POS tagging is similar to hard clustering, as there is only one tag per word. In this case, the entropy $H(\mathcal{C}) = -\sum_{k=1}^{K} P(C_k) \log P(C_k)$ of the random variable $\mathcal{C}$, corresponding to the choice of the cluster C_k among K clusters, is readily computable (see Meila, 2003). However, in the case of a morphological analysis, there are several morpheme labels per word. The random variable is thus the binary vector $M = (b_1, \ldots, b_n)$, where $b_i = 1$ if morpheme i occurs in the word. If we consider calculating the entropy, $H(\mathcal{M}) = -\sum_M P(M) \log P(M)$, there are at least two problems. First, it is hard to estimate $P(M)$: independence of the morphemes is not likely to be a good assumption, but there is hardly enough data to do anything else. Second, one has to sum over 2^n choices of M, which is impossible in practice for any morpheme lexicon of a reasonable size.

Although there does not seem to be any simple way to apply the information-theoretic measures to the evaluation of full analyses, they can still be useful for subproblems of the morphology learning. For example, Chan (2006) evaluates signatures based on POS tags using entropy-based measures: *POS fragmentation* measures the entropy of the signatures conditioned on the distribution of POS tags, and *signature impurity* measures the entropy of POS tags conditioned on the distribution of the suffixes in the signatures.

3.1.6. *Other Direct Evaluations*

In some cases, a method for morphology learning is developed as a model of language acquisition (see, e.g., Chan, 2008). Application evaluations or evaluations based on linguistic reference may not be as relevant for this goal as for others. Lignos *et al.* (2010b) evaluate their model by applying it to child-directed data and manually comparing its learning process to the research in child language acquisition. Another option is to do direct comparison using behavioral studies. For example, Lim *et al.* (2005) study a trie structure for storing Korean words, and find that the search times correlate to three properties of words and non-words (frequency, length, and non-words similarity to a correct word) in a similar manner as human reaction times. In a recent work, Virpioja *et al.* (2011) study how an unsupervised probabilistic model can predict reaction times for Finnish nouns in a lexical decision task. These can be considered as direct evaluations, although the external “reference” is not an analysis by linguists but something measured from human test subjects.

3.2. *Application Evaluations*

The most important NLP application for morphological analyzers has so far been information retrieval, where people often want to find documents including given words regardless whether they are inflected or parts of compound words. However, the problem of a huge vocabulary of words in morphologically rich languages concerns directly all applications that need a statistical model for the language. Common examples are speech recognition, which only needs to deal with the surface forms of the words, and statistical machine translation.

3.2.1. *Information Retrieval*

A useful comparison of unsupervised morphological analysis methods is how well they perform in an information retrieval task. Morphological analysis is needed, since all matching documents should be retrieved irrespective of which word forms are used to describe the contents of the documents and the queries. The evaluation is carried out simply by replacing the inflected words in the corpus and the queries by the suggested morpheme analyses. The performance of the unsupervised algorithms can be compared to doing no analysis at all or to the performance of rule-based morphological analyzers or stemming.

Evaluating unsupervised segmentation algorithms in terms of IR performance has been done already by Hafer and Weiss (1974). Segmentation based on LSV yielded similar IR performance for English than stemming. Naturally, the effect of different morphological analysis or stemming strategies have been extensively studied in the field of IR, but usually focused only on language specific methods. Alkula (2001) compared IR performance on Finnish using different morphological analyzers and stemmers. Best performance was achieved by using base forms.

Algorithms should be compared using multiple languages, because the importance of morphological analysis for IR depends on the language. Pirkola (2001) presents a morphological classification of languages from the IR point of view. For morphologically simple languages such as English, simply removing affixes (stemming) is often enough. For morphologically more complex languages such as Finnish, morphological analysis is needed to turn word forms to their base forms (*lemmatization*) and to split compound words into their parts (*decompounding*). Successful lemmatization conflates word forms with similar meanings and separates ones with different meanings. Thus lemmatization improves recall without hurting precision. In decompounding, recall is also improved but precision may suffer.

For the algorithms that only segment the words, the IR performance depends entirely on how aggressive the segmenting is. If the stem is too short, words that should remain distinct are conflated. If the stem is too long, words that should be merged together remain distinct. The former is called *overstemming* and causes precision to drop. The latter is called *understemming* and causes recall to drop. Due to non-concatenative processes such as allomorphy, it is not even theoretically possible to always find segment boundaries that avoid under- and overstemming issues. For best IR performance, the unsupervised algorithms should also try to learn these phenomena.

In the Morpho Challenge series, IR evaluations were introduced in 2007 (Kurimo *et al.*, 2007) using English, Finnish and German. To accurately measure the effect of the morphological analysis on IR, the number of other variables will have to be minimized. For example, different term weighting approaches may give different results depending on which morpheme segmentation or analysis method is used. When TFIDF and Okapi BM25 weighting approaches were tested in Morpho Challenge 2007, it was noted that Okapi BM25 suffers greatly if the corpus has a large number of very frequent terms. Frequent terms are introduced by methods that separate suffix morphs. If the suffix morphs were tagged, they could be removed, but most compared methods did not tag the morphs. Simply removing terms with a corpus frequency higher than some threshold improved Okapi BM25 results to a clearly higher level than TFIDF weighting for all algorithms. With this method of generating automatic morpheme stop lists, all algorithms could be treated equally.

The evaluation criterion for the compared algorithms is the obtained *Mean Average Precision* (MAP) in the IR task. For each query, the ranked list of documents returned by the system is compared to the known relevant documents. Let $\mathrm{rel}(k)$ equal 1 if the document at rank k is relevant and 0 if it is not. Precision at rank k (Pre(k)) is the proportion of relevant documents among the k topmost documents:

$$\mathrm{Pre}(k) = \frac{1}{k}\sum_{i=1}^{k}\mathrm{rel}(k). \qquad [16]$$

Let R be the number of relevant documents for a query. Average Precision (AP) for the query is the average of precisions at ranks that have a relevant document:

$$\mathrm{AP} = \frac{1}{R}\sum_k \mathrm{Pre}(k)\,\mathrm{rel}(k). \qquad [17]$$

MAP is the mean of Average Precisions over all queries.

A complication for analyzing the results is the fact that it is hard to achieve statistically significant results with the limited number of queries available. This is a known problem in the field of IR. However, the performance of the algorithms across languages provides a useful comparison of their success. The results (Kurimo *et al.*, 2010a) showed that language specific reference methods give the best results, but the best unsupervised algorithms are almost at par and the differences are not significant.

3.2.2. *Speech Recognition*

An essential part of any large vocabulary speech recognition system is a language model that can provide a probabilistic ranking for all word sequences that the recognizer proposes. In morphologically rich languages, the estimation and use of such a statistical model is very challenging and sets remarkable requirements for training data and computational resources. This problem was one of the main motivations to develop the Morfessor Baseline algorithm (Creutz and Lagus, 2002) ten years ago, which enabled the construction of the unlimited vocabulary dictation system (Hirsimäki *et al.*, 2006) with morph-based language models. The completion of this system, and a corresponding one in Turkish, made it then possible to run the first Morpho Challenge (Kurimo *et al.*, 2006), where all submitted algorithms for unsupervised morphemes could be tested in state-of-the-art speech recognition tasks.

The Morpho Challenge 2005 evaluation (Kurimo *et al.*, 2006) was successful in pointing out unsuitable morphemes, but it failed to provide statistically significant differences in recognition error rate for the top algorithms. Furthermore, large scale speech recognition evaluations required a substantial amount of work. For each compared morpheme lexicon, we trained a new language model from all the available text data in that language, re-optimized the whole recognition system in the development data, and recognized the test speech that was long enough to provide a statistically meaningful error rate. Thus, these evaluation metrics were not computed again in later Morpho Challenge evaluations, even though speech recognition in morphologically rich languages continues to be one of the main applications for large-scale morphological analysis (Hirsimäki *et al.*, 2009).

3.2.3. *Machine Translation*

Machine learning approaches to morphology are not very relevant for traditional rule-based machine translation systems, as they, in any case, require hand-crafted linguistic rules. The situation is different for statistical machine translation (SMT), for which the ideal situation would be that the same models worked for many languages. For example, Virpioja *et al.* (2007) proposed using an unsupervised method as a language-independent tool for morphological preprocessing before training phrase-based SMT systems.

There is a large amount of work for dealing with morphological analysis, decomposition or stemming in SMT (e.g. Nießen and Ney, 2004; Yang and Kirchhoff, 2006; Oflazer and El-Kahlout, 2007), but the viewpoint has been on how the information provided by a morphological analyzer (or a morphologically annotated corpus) should be applied in the SMT framework. Comparison of different methods for morphological analysis using the same SMT system has so far been done only in the recent Morpho Challenges (Kurimo *et al.*, 2010b; Kurimo *et al.*, 2010c).

The SMT evaluations in Morpho Challenges have included two tasks, Finnish to English and German to English, and the morphological analyses have been applied only to the source language. The main problem in the evaluation has been that using just the words provides often the best results. This is hardly surprising considering that the SMT models have been designed for word-based translation, and that the target sentences contain full words. To avoid that the submissions with the least amount of segmentation would get the best results, the evaluation setup has included combining the results with those obtained using a plain word-based model. Still, only a few algorithms have given statistically significant improvements over the word-based translation. As morphologically rich languages pose a major problem to the results of SMT, it is likely that translation systems that can better incorporate morphological analyses will be available sooner or later.

4. Experiments

The Morpho Challenge competitions have provided a database that consists of the results of about fifty algorithms for unsupervised and semi-supervised learning of morphology, evaluated for several tasks and languages. In the optimal case, the input data for training the algorithms and the output data used in the evaluations would have been the same each year, and every algorithm would have participated in every task. Unfortunately, that is not the case, and the database is somewhat sparse. First, we excluded the results of Morpho Challenge 2005, where the data sets were significantly different from those in the following Challenges. Second, we excluded the Arabic language, which was included in two Challenges (2008 and 2009), but only for the linguistic evaluation and using different data sets each time. For the remaining languages and tasks, Table 2 shows the number of evaluated methods. We have no reference segmentations for German, so boundary evaluations could not be applied to it.

Table 2. *The number of methods evaluated in different tasks and languages. "Segmentation" refers to the methods that could be evaluated by measuring precision and recall of the morph boundaries.*

Evaluation	*Number of methods*			
	English	*Finnish*	*German*	*Turkish*
Linguistic evaluation	49	42	39	45
– Segmentation	20	18	0	20
Information retrieval	36	31	25	0
Statistical machine translation	0	22	13	0

4.1. *Linguistic Evaluations*

As evaluation methods, we tested the following isomorphic evaluations: Morpho Challenge evaluation method from 2009 and 2010 (MC), EMMA (Spiegler and Monson, 2010), the modified EMMA using two one-to-many mappings as described in section 3.1.4 (EMMA-2), and the new co-occurrence based methods described in section 3.1.3 (CoMMA). Among these, only EMMA is a hard isomorphic evaluation, while the rest are soft isomorphic evaluations. As the boundary evaluation described in section 3.1.2 is a very simple and popular method, we included it as a baseline. In order to make it compatible to alternative analyses present in the data, we used a similar approach to CoMMA-S and matched them using the Hungarian algorithm to get maximal average F-score. This method is referred to as "BPR". Table 3 shows a comparison of the evaluations, including the type of evaluation, how alternative analyses are handled, and whether isolated words that have no common morphemes to other words in the evaluation set are excluded.

Table 3. *Methods for linguistic evaluation: the type of evaluation, treatment of alternative analyses, and whether isolated words are included.*

Name	*Evaluation type*	*Alternatives*	*Isolated words*
BPR	Boundary positions	Best match	Included
MC	Co-occurrence	Best single pair	Excluded
EMMA	Assignment (1-1)	Best match	Included
EMMA-2	Assignment (M-1 / 1-M)	Best Pre / Rec	Included
CoMMA-B0	Co-occurrence	Reduced to max	Excluded
CoMMA-B1	Co-occurrence	Reduced to max	Included
CoMMA-S0	Co-occurrence	Best match	Excluded
CoMMA-S1	Co-occurrence	Best match	Included

The reference analyses used in the linguistic evaluations were the same as in the Morpho Challenges 2007-2010 (Kurimo *et al.*, 2008; Kurimo *et al.*, 2009; Kurimo *et al.*, 2010c; Kurimo *et al.*, 2010b). The English and German gold standards were based on the CELEX database (Baayen *et al.*, 1995). The Finnish gold standard was based on the FINTWOL analyzer from Lingsoft, Inc., that applies the two-level morphology model by Koskenniemi (1983). The English and Finnish reference analyses were transformed to segmentations via Hutmegs (Creutz and Lindén, 2004). The Turkish reference analyses, including segmentation, were obtained from a morphological parser developed at Boğaziçi University.

The Morpho Challenge evaluation differs from the other methods in that it first requires sampling of the word pairs. We applied the same word pair lists as in the Morpho Challenges. They included 10,000 (English), 200,000 (Finnish), 50,000 (German), and 50,000 (Turkish) focus words from the gold standards. For the other evaluation methods, we collected all the words present in the word pair lists, and sampled ten random subsets of 1,000 word forms. For each evaluation method and evaluated algorithm, we calculated precision, recall and F-score for each subset, and then took the average. The sets of average scores were used for calculating Spearman's rank correlation coefficients to the results of the application evaluations.

In addition, to study the computation time and stability of the evaluation methods with respect to the size of the evaluation data, we sampled ten sets of 100, 300, 1000, 3,000, and 10,000 word forms from the English and Finnish test sets. Due to the long evaluation times with the larger sets, we used these sets to evaluate only one algorithm, Morfessor Baseline.

4.2. *Information Retrieval Tasks*

Our information retrieval tasks were the same as in Morpho Challenge 2010 (Kurimo *et al.*, 2010b). Three languages were used: English, German and Finnish. Test corpora, queries and relevance assessments were provided by Cross-Language Evaluation Forum (CLEF) (Agirre *et al.*, 2008). To evaluate the algorithms, the IR tasks were run after replacing all word forms in the corpora and the queries by the submitted analyses. Success was measured in terms of Mean Average Precision (MAP). The evaluations were carried out with the Lemur Toolkit (Ogilvie and Callan, 2002) using Okapi BM25 ranking with default parameter values. For each submission, a stop list was generated, since Okapi BM25 suffers if the corpus contains terms that are very common. Any term that has a collection frequency higher than 75,000 (Finnish) or 150,000 (German and English) was excluded from indexing.

The information retrieval task has been repeated four times in Morpho Challenges 2007-2010. The data, setup and methods for the task have remained the same. However, a number of issues have been detected and fixed over the years. In the 2007 challenge, a part of the evaluation corpus for German was missing, making the task easier and thus the results comparably higher. On all languages, some fixes were made to corpus preprocessing and word list generation that had small effects on the results. For this paper, all algorithms from previous years were re-evaluated to make the results comparable to the 2010 results.

4.3. *Statistical Machine Translation Tasks*

We used the machine translation evaluation from Morpho Challenge 2010 (Kurimo *et al.*, 2010b). The translation was done from a morphologically complex source language (here Finnish and German) to English. The words of the source language were replaced by their morpheme analyses before training the translation models. The morpheme-based models were combined to a standard word-based model by generating n-best lists of translation hypotheses from both models, and finding the best overall translation with the Minimum Bayes Risk (MBR) decoding (Kumar and Byrne, 2004). A state-of-the-art phrase-based SMT system, Moses (Koehn *et al.*, 2007), was used for training the translation models and generating the n-best lists. As an evaluation measure, we used the BLEU metric (Papineni *et al.*, 2002).

The Europarl corpus (Koehn, 2005) was used for training and testing the SMT systems. It was divided into three subsets: training set for training the models (about 300,000 sentences), development set for tuning the model parameters (about 3,000 sentences), and test set for evaluating the translations (about 3,000 sentences). The word lists given for participants for learning morphology included all the word forms in the Europarl corpus, including the test data.[6] The data sets and the evaluation setting were almost the same as in Morpho Challenge 2009 (Kurimo *et al.*, 2010c), but there was one change in 2010. As the alignment tool used in training the SMT system has a limitation of 100 tokens per sentence, all the sentences that had more than 100 letters were discarded. This way, all the systems had the same amount of training data regardless of the number of morphemes found. To get comparable results, we re-evaluated all the algorithms from the machine translation competition of 2009.[7]

4.4. *Algorithms in the Evaluation*

In this section, the main algorithms in Morpho Challenges 2005-2010 are described very briefly. Including all the variants of these, there were more than 50 submissions which are presented in detail in the following publications.

Bernhard [1, 2] 2007 (Bernhard, 2008) first extracts a list of the most likely prefixes and suffixes and then generates alternative segmentations for the word forms. The best ones are selected based on cost functions that favour most frequent analysis and some basic morphotactics.

[6] A more realistic setting would be that the model learned on the training data would be applied to analyze the tuning and test data. However, the practical arrangements for such a setting would be more complicated.

[7] However, a slight difference remains between the results from 2009 and 2010 algorithms. In 2009, the data set for the SMT competition was separate from the other data sets, while in 2010 there was one combined data set. The participants of Challenge 2009 were allowed to use also other data sets than the SMT set, but some may not have done so.

Bordag [5, 5a] 2007 (Bordag, 2008) applies iterative letter successor variety (LSV) and clustering of morphs into morphemes.

Can [1, 2] 2009 (Can and Manandhar, 2010) use unsupervised part-of-speech tagging as an initial step to find morphological paradigms. This has so far been the only approach to exploit the context information of the words.

DEAP [MDL-CAT, MDL-NOCAT, PROB-CAT, PROB-NOCAT] 2010 (Spiegler *et al.*, 2010a) is a supervised algorithm using deductive-abductive parsing with a context-free grammar. The best hypothesis from abduced parses is selected either using a probabilistic or MDL-inspired criterion. For the NOCAT versions, the morpheme labels returned by the parsing algorithms were removed, thus returning only a segmentation, and for the CAT versions, they were kept.

Lignos 2009, 2010 (Lignos *et al.*, 2010a; Lignos, 2010) is based on the observation that the derivation of the inflected forms can be modeled as transformations. The best transformations can be found by optimizing the simplicity and frequency. The submissions in 2010 included three variants, **Base Inference**, **Aggressive Compounding**, and **Iterative Compounding**.

McNamee [3, 4, 5] 2007 (McNamee and Mayfield, 2007) extracts all the letter n-grams in the words, and for each word selects the n-gram that occurred the least number of times in total. The different versions apply different n-gram lengths (3, 4, and 5). This method was intended mainly for the IR task.

MetaMorph 2009 (Tchoukalov *et al.*, 2010) applies multiple sequence analysis (MSA), which are popular in biological sequence processing, to the problem of learning morphology. The approach is problematic for large sets of word forms, but more useful for smaller sets of orthographically related words.

Morfessor Baseline (Creutz and Lagus, 2002; Creutz and Lagus, 2005b) is a public baseline algorithm based on jointly minimizing the size of the morph codebook and the encoded size of all the word forms using the minimum description length (MDL) cost function.

Morfessor Categories-MAP (Creutz and Lagus, 2005a) is a extension of the Morfessor Baseline method, where hidden Markov models are used to incorporate morphotactic categories. The structure is optimized using maximum a posteriori (MAP) estimation. **Morfessor Categories-MAP 2007** and **2008** are submissions by Monson *et al.* (2008) and Monson *et al.* (2009) using the same method.

Allomorfessor 2008, 2009 (Kohonen *et al.*, 2009; Virpioja *et al.*, 2010) is an extension of Morfessor Baseline, where stem allomorphy is modeled using string mutations that modify the letters close to the morpheme boundary.

Morfessor [U+W, S+W, S+W+L] 2010 (Kohonen *et al.*, 2010a; Kohonen *et al.*, 2010b) are semi-supervised versions of Morfessor Baseline. **U+W** uses supervision only to find a suitable weight for the data likelihood, whereas **S+W** also uses the known segmentations to guide the search algorithm. In **S+W+L**, a hidden Markov model is trained to label the segments according to those present in the annotated data.

MorphAcq 2010 (Nicolas *et al.*, 2010) is an unsupervised approach to find a set of morphological rules, i.e., transformations that convert a stem into a related lexical form. The method applies several strategies that first find pairs of candidate affixes and then build morphological rules. Special attention is paid to frequency-related phenomena.

MorphoNet 2009 (Bernhard, 2010) is based on finding community structure from a lexical network. The lexical network is constructed by learning transformation rules based on graphical similarities between words.

ParaMor 2007-2010 (Monson *et al.*, 2008; Monson *et al.*, 2009; Monson *et al.*, 2010) applies an unsupervised model for inflection rules and suffixation for the stems by building linguistically motivated paradigms. **ParaMor-Morfessor** (2007, 2008) combines ParaMor with Morfessor Categories-MAP by giving the analyses of the both methods as alternatives. **ParaMor-Morfessor Union** (2010) combines ParaMor and Morfessor by taking union of the segmentation points. **ParaMor Mimic** and **ParaMor-Morfessor Mimic** (2010) are supervised probabilistic models trained on the results of the unsupervised algorithms.

Promodes 2009-2010 (Spiegler *et al.*, 2010b; Spiegler *et al.*, 2010c) presents a probabilistic generative model that considers morpheme boundaries as hidden variables and includes probabilities for letter transitions within morphemes. In both years, there were three different versions, one being a combination of the other two.

RALI-ANA and **RALI-COF 2009** (Lavallée and Langlais, 2010) identify transformations between word forms using formal analogy, i.e., relations of four forms such as *reader* is to *doer* as *reading* is to *doing*. RALI-ANA is a pure analogical approach, while RALI-COF applies related but more general cofactor rules instead.

RePortS 2007 (Keshava and Pitler, 2006) uses simple LSV-type criteria based on two letter n-gram models that predict forward and backward to score potential prefixes and suffixes. Combinations of other affixes are pruned from the candidate list, and the final segmentation points determinated using the letter models.

UNGRADE 2009 (Golénia *et al.*, 2010a) aggregates two types of information: stem candidates found using a MDL-type criterion, and affix candidates found using a graph-based, LSV-type approach.

MAGIP 2010 (Golénia *et al.*, 2010b) is a supervised approach that creates a morpheme graph similar as in UNGRADE, but trains it on known segmentations. Mixed-integer programming is applied to select the best parse of an unseen word from the set of parses generated from the graph.

Zeman [1, 3] 2007, 2008 (Zeman, 2008; Zeman, 2009) uses a heuristic algorithm to find paradigms assuming that there is only one stem and suffix per word. In 2008, the method was extended to search also for prefixes using two different approaches.

4.5. *Results*

The evaluation results for the participating algorithms are presented in the respective Morpho Challenge overview articles, so we do not concentrate on those in this article. Table 4 shows selected results for some of the algorithms for the Finnish tasks. The full result tables for all algorithms, languages and evaluations are found at `http://research.ics.tkk.fi/events/morphochallenge/`. The importance of the analysis of the evaluation metrics is clear from Table 4: the differences in the evaluations result in different orders for the algorithms. There are at least two reasons for the differences of direct evaluations: some measure different things (morph boundaries or morpheme sets), and some measure the same thing but in a different manner (using co-occurrences or soft or hard assignment).

Table 4. *Selected results for some of the evaluated methods for Finnish. T is the type of the algorithm: semi-supervised (S), unsupervised (U), or unsupervised with supervised parameter tuning (P). #a/w is the average number of analyses per word, #m/w is the the average number of morphemes per word, and #lex is the size of the morpheme lexicon. The best score for each metric is in bold.*

Method	*T*	*#a/w*	*#m/w*	*#lex*	*MC*	*EMMA*	*IR*	*SMT*
				×1k	*F*	*F*	*MAP*	*BLEU*
Allomorfessor 2009	U	1.00	2.46	70	32.44	58.25	45.68	**26.80**
Bernhard 2 2007	U	1.00	3.89	88	52.45	61.11	**49.07**	-
Bordag 5a 2007	U	1.00	2.84	515	39.56	58.41	42.83	-
DEAP MDL-CAT	S	5.31	3.23	2,549	61.67	31.66	37.25	25.90
DEAP MDL-NOCAT	S	3.29	3.41	1,753	**62.52**	40.95	41.60	25.62
Lignos Base Inference	U	1.00	2.58	560	37.35	65.88	41.51	26.19
Morfessor Baseline	U	1.00	2.17	176	24.83	58.44	42.35	26.65
Morfessor CatMAP	U	1.00	2.88	239	43.16	61.14	47.54	26.34
Morfessor S+W	S	1.00	4.20	14	56.38	62.08	47.50	25.94
Morfessor S+W+L	S	1.00	4.27	19	60.76	**71.19**	44.65	25.82
MorphoNet 2009	U	1.00	2.53	985	33.34	56.22	38.75	25.56
ParaMor 2008	U	1.00	2.62	1,124	42.93	55.58	38.28	-
ParaMor Mimic	P	1.00	3.30	1,149	43.57	55.46	39.05	25.53
ParaMor-Morf. Mimic	P	1.00	4.24	324	48.38	57.69	44.46	25.54
ParaMor-Morf. Union	P	1.00	4.02	215	49.39	55.79	47.13	25.44
Promodes 2010	P	1.00	5.46	236	44.31	51.18	37.21	25.64
RALI-COF	U	1.00	2.39	723	38.81	63.94	-	-

4.5.1. *Correlation to Application Evaluations*

The upper part of Figure 4 shows the correlations between the results of linguistic evaluation methods and the IR tasks. EMMA provides high correlations (≥ 0.7) for English and Finnish and moderate (≥ 0.5) for German. EMMA-2 gives slightly lower correlation for German, but is otherwise close. The MC evaluation is among the worst both in English and Finnish, but, surprisingly, provides the highest correlation for German. Among the CoMMA methods, CoMMA-S gives higher correlations than

CoMMA-B. Including isolated words improves correlation for English and Finnish, but decreases for German. For the SMT tasks (lower part of Figure 4), only EMMA shows a positive correlation for both languages, and EMMA-2 and CoMMA-B1 for Finnish, while all the others have negative correlations.

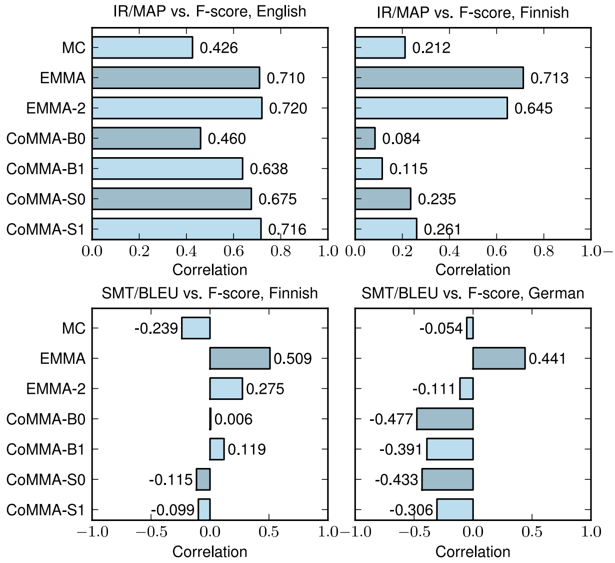

Figure 4. *Spearman's rank correlations between the F-scores of the linguistic evaluation methods and the scores of information retrieval and statistical machine translation evaluations.*

For co-occurrence based metrics, it was clear that the balanced F-score would not give the best correlations to the application evaluations, as the optimal balance between precision and recall is likely to depend on the application and language. To study this effect, we calculated the correlations for a set of F_β-scores with different weights β. The results are plotted in Figure 5. For the CoMMA methods, as well as MC, the optimal β is always below one, emphasizing precision. CoMMA-B0 and S0 require usually more weighting than B1 and S1 to get as high correlations. For EMMA and EMMA-2, the optimal β is towards recall with the IR tasks. Peculiarly, the recall of EMMA gives the highest overall correlation for the German IR. For Finnish SMT, all methods give the best correlation just for precision. It seems that the most of the algorithms in the database made too recall-oriented analyses for this task.

With weighted F_β, EMMA still provides the highest correlations for the IR tasks, and EMMA-2 is close, but the best correlations for the co-occurrence based metrics are much closer. For Finnish SMT, MC and CoMMA-B variants give higher correlation for the precision than the others, and for German SMT, all the peaks are very close. Figure 6 shows a similar plot for the English and Finnish IR tasks, but including

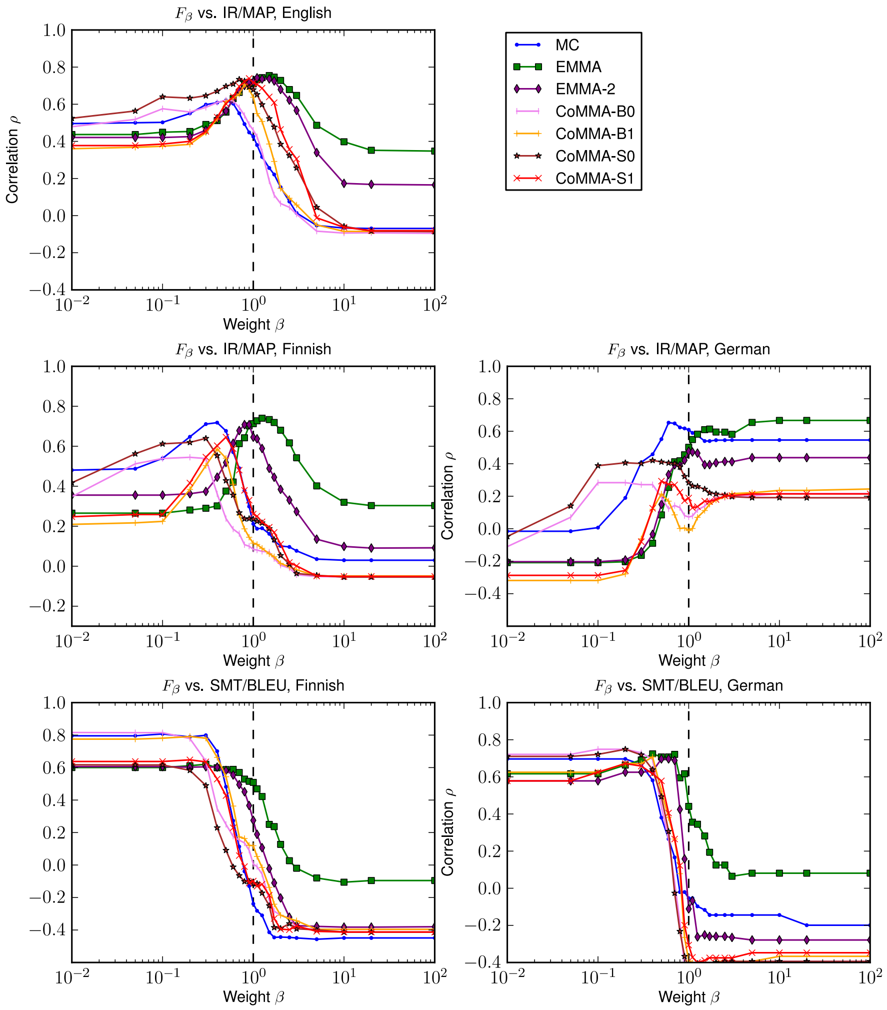

Figure 5. *Spearman's rank correlations between the results of the application evaluations and weighted F_β-scores with varying β.*

only the algorithms that return a segmentation so that the boundary evaluation (BPR) can be included. Note that for this subset of the algorithms, the overall level of correlations is higher. BPR gives the third highest numbers after EMMA and EMMA-2 for both languages.

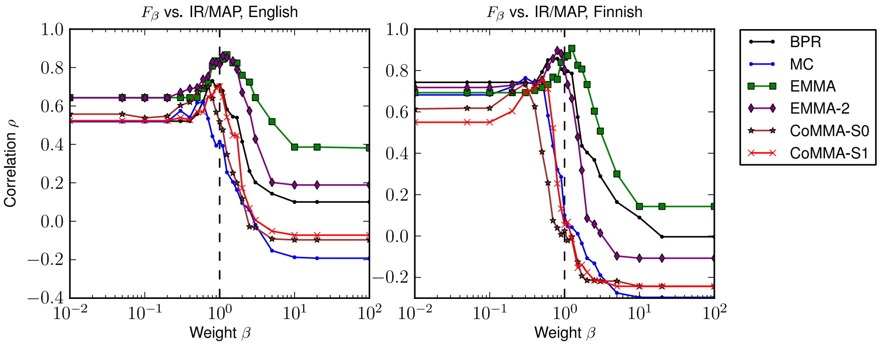

Figure 6. *Spearman's rank correlations between the results of the IR evaluations and weighted F_β-scores with varying β. Only segmentation algorithms are included in the evaluation.*

4.5.2. *Correlation to Boundary Evaluation*

For languages that have mostly concatenative morphology, it is useful to know how well the isomorphic evaluations correlate to boundary evaluation. In many cases, a linguistic reference does not include a segmentation but only morpheme labels, while the evaluated algorithm does only segmentation. If the correlation is high, it is possible to substitute the isomorphic evaluation for the boundary evaluation. Figure 7 shows correlations between the F-scores of BPR and the isomorphic evaluations. The results vary over the languages, and only EMMA and EMMA-2 provide high correlations in all of them. CoMMA-B1 and CoMMA-S1 have the best correlations in English, and MC in Turkish, but all of them have only moderate correlation in Finnish.

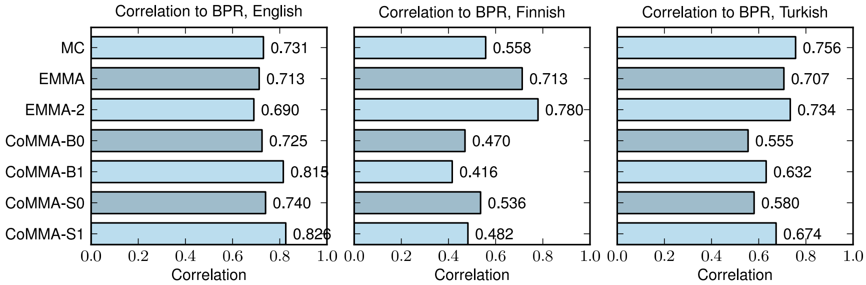

Figure 7. *Spearman's rank correlations of the F-scores of the isomorphic evaluation methods and the BPR boundary evaluation.*

4.5.3. *Robustness*

To test robustness of the evaluations with respect to gaming, we used the tests introduced by Spiegler and Monson (2010).

Ambiguity hijacking test addresses how the evaluation method deals with alternatives in the predicted analyses. As some of the words are ambiguous regarding their morphological analysis, the evaluation methods should allow the alternatives. However, providing two alternative analyses for a non-ambiguous word should not give higher score than providing a reasonable combined analysis or just the better one. For example, *ParaMor-Morfessor*, which simply lists the analyses of ParaMor and Morfessor Categories-MAP as two alternatives, should not outperform *ParaMor-Morfessor Union*, which combines the morpheme boundary predictions as a single analysis. Figure 8 shows that MC and CoMMA-B give higher F-scores to ParaMor-Morfessor than to ParaMor-Morfessor Union, while CoMMA-S and EMMA-2 are as robust as EMMA in this respect.

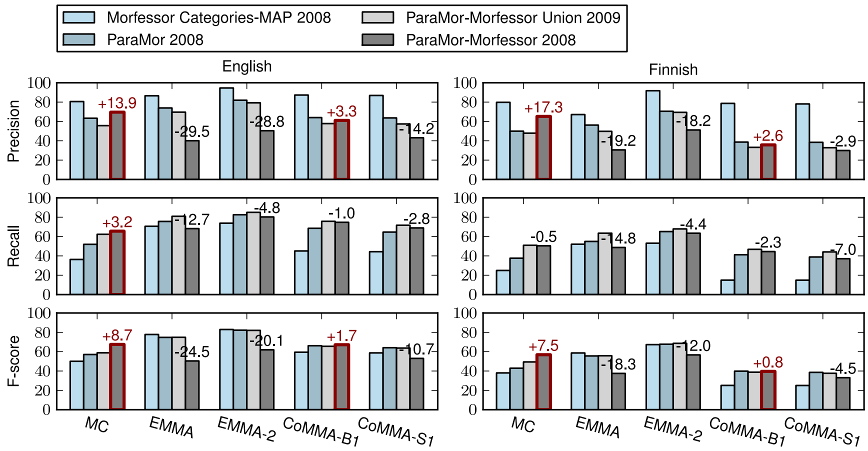

Figure 8. *Gaming with ambiguity hijacking on English and Finnish: ParaMor-Morfessor returns ParaMor and Morfessor Categories-MAP as two alternatives, whereas ParaMor-Morfessor Union combines the two predictions into a single analysis. The number above ParaMor-Morfessor 2008 shows the absolute difference to ParaMor-Morfessor Union. CoMMA-B0 gives similar results to B1 and S0 similar results to S1.*

Shared morpheme padding test addresses the vulnerability of the evaluations to an artificial modification of the analysis. A unique bogus morpheme is added to each predicted analysis. For methods based on co-occurrence analysis, this means adding an additional edge between each word. As expected, the results in Table 5 show that the recall scores are clearly increased and precision scores decreased for the MC and CoMMA methods. For those languages where high recall was hard to obtain (Finnish and Turkish), this improves the F-score, while for those where the recall was initially high (English and German), the F-score decreases. EMMA-2 is almost as robust as the original EMMA, showing only small changes in the scores.

Table 5. *Gaming with shared morpheme padding: average and standard deviations of the ratio of padded to original scores for the evaluation methods.*

Lang.	*Precision*	*Recall*	*F-score*	*Precision*	*Recall*	*F-score*
		MC evaluation				
English	0.36±0.08	2.02±0.66	0.63±0.10			
Finnish	0.57±0.08	3.07±2.46	1.19±0.68			
German	0.43±0.08	2.90±1.45	0.84±0.16			
Turkish	0.58±0.09	2.95±1.65	1.19±0.37			
		EMMA evaluation			EMMA-2 evaluation	
English	0.73±0.15	1.05±0.08	0.86±0.12	0.76±0.07	1.28±0.10	0.96±0.03
Finnish	0.87±0.19	1.12±0.10	0.99±0.14	0.86±0.05	1.62±0.25	1.18±0.09
German	0.80±0.17	1.09±0.08	0.94±0.11	0.79±0.05	1.52±0.22	1.07±0.08
Turkish	0.85±0.08	1.07±0.04	0.97±0.05	0.85±0.05	1.76±0.32	1.29±0.15
		CoMMA-B0 evaluation			CoMMA-B1 evaluation	
English	0.15±0.10	2.24±0.81	0.31±0.13	0.12±0.04	1.86±0.46	0.23±0.06
Finnish	0.51±0.14	6.46±7.88	1.76±1.74	0.44±0.11	5.03±4.44	1.34±0.89
German	0.28±0.12	3.35±2.91	0.59±0.35	0.21±0.05	3.07±2.10	0.49±0.24
Turkish	0.48±0.15	5.65±4.51	1.76±1.14	0.43±0.12	5.91±4.58	1.46±0.88
		CoMMA-S0 evaluation			CoMMA-S1 evaluation	
English	0.15±0.10	2.24±0.81	0.31±0.13	0.16±0.17	1.79±0.46	0.28±0.16
Finnish	0.51±0.14	6.46±7.88	1.76±1.74	0.46±0.14	4.67±3.92	1.34±0.83
German	0.28±0.12	3.35±2.91	0.59±0.35	0.24±0.16	2.94±2.01	0.52±0.25
Turkish	0.48±0.15	5.65±4.51	1.76±1.14	0.45±0.15	4.57±3.02	1.46±0.74

4.5.4. *Interpretability*

Interpretability of an evaluation method, as defined by Spiegler and Monson (2010), concerns how the evaluation results can be used for identifying the strengths and weaknesses of the predicted analyses. The F-scores of all the discussed evaluation methods are readily interpretable in the sense that they measure well-defined properties of the predicted analyses: EMMA and EMMA-2 measure how well the predicted morphemes can be matched to reference morphemes, while MC and CoMMA measure whether the words have the correct number of shared morphemes in the predicted analysis. However, the evaluations can also provide additional information on the evaluated analyses.

EMMA has the benefit of providing a mapping between the predicted and the reference morphemes. This is useful especially for human inspection of the results as it helps qualitative evaluation. This applies also to EMMA-2, but as it provides two many-to-one mappings that often have some obscure mappings for the morphemes that occur only once, they are not as easy to utilize as in EMMA.

Instead of studying individual morphemes or analyses, sometimes a more general view on the result is more useful. One question is whether the precision and recall of the evaluation method can provide useful information. As explained in section 3.1.3, low recall in co-occurrence based metrics should mean that you are missing some co-occurrences (e.g., not segmenting enough or not joining allomorphs) and low precision that you have spurious co-occurrences (e.g., segmenting too much or having the same label for syncretic morphemes). In contrast, one-to-one matching gives neither good precision nor recall if the number of predicted morphemes is wrong.

To study this experimentally, we trained the Morfessor Baseline algorithm with different likelihood weights (see Kohonen *et al.*, 2010a), thus controlling the amount of segmentation. Then we calculated precision and recall for the results using the different evaluation methods, shown in Figure 9. The points in upper-left corner correspond to models that severely undersegment, and the amount of segmentation increases by each point. All co-occurrence based methods (including the CoMMA variants not in the figure), boundary evaluations, and EMMA-2 have recall and precision that consistently decrease and increase, respectively, when the words are segmented more. If the evaluated algorithm gets, for example, Pre $= 0.45$ and Rec $= 0.5$, it indicates that the analyses are balanced in that the amount of missing co-occurrences and the amount of spurious co-occurrences are about the same. With EMMA, recall starts to decrease after a certain point, obstructing this kind of interpretations.

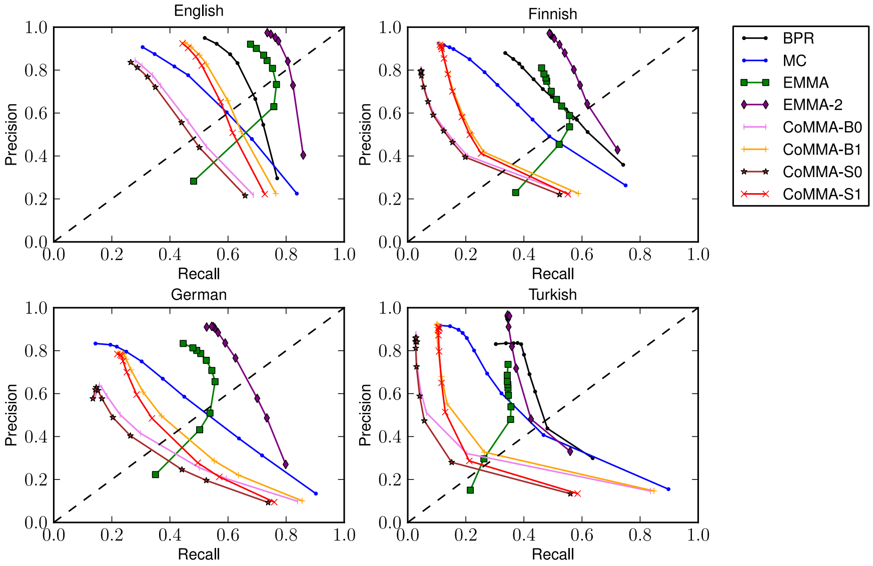

Figure 9. *Precision-recall curves of the evaluation methods for Morfessor Baseline models with varying amount of segmentation. The points in the upper-left corner correspond to models that resulted in fewer segmentations.*

4.5.5. *Computation Time*

Figure 10 shows the average computation times of the evaluation methods for evaluating Morfessor Baseline using evaluation sets of varying size. The MC evaluation is excluded, as its approach based on random sampling is very different from the others. The boundary evaluation (BPR) is very fast, having in practice a linear complexity. All CoMMA variants show polynomial growth of the same order (linear and same slope in the log-log scale). For EMMA, 16 GB of memory are not enough for the 3,000-word sets, so we had to stop at 2,000 words. The growth of the computation time is faster than with CoMMA, potentially exponential. EMMA-2 was very fast for the tested evaluation sets, but the super-linear trend in the log-log scale indicates exponential growth for it, too. The exponential growth in EMMA and EMMA-2 is an implementation issue, related to using the integer linear programming for morpheme assignment (only in EMMA) and for matching the alternatives (in both).

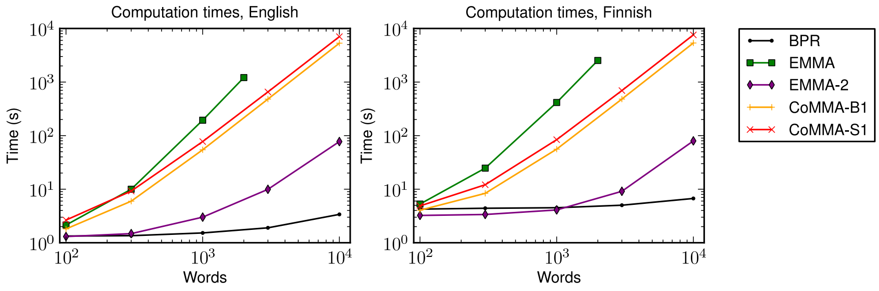

Figure 10. *Computation times of different evaluation methods with respect to the size of the evaluation data (100-10,000 words) for English and Finnish. Both the time and the number of words are shown in logarithmic scale. The evaluated method is Morfessor Baseline.*

4.5.6. *Stability for Evaluation Data Variations*

In order to study the stability of the evaluation methods with respect to the size of the evaluation data, we calculated precision, recall and F-score for Morfessor Baseline using the sets from 100 to 10,000 words. The means and standard deviations of the results are plotted in Figure 11. Unsurprisingly, boundary evaluation is a very stable method with respect to the size of the data. The MC evaluation shows more variation: for Finnish, all scores are underestimated with small data sets, while for English, they are first overestimated and then underestimated. EMMA and EMMA-2 give smaller standard deviations than the other methods, but they clearly overestimate the scores with small data sets. For CoMMA, S0 and B0 as well as S1 and B1 give similar results, so only the formers are included. Variants that exclude isolated words show a similar pattern as the MC evaluation, but the changes are smaller (in particular for the recall in Finnish). Variants that include isolated words overestimate the scores with small data sets, but in contrast to EMMA and EMMA-2, the changes get smaller as the data size grows.

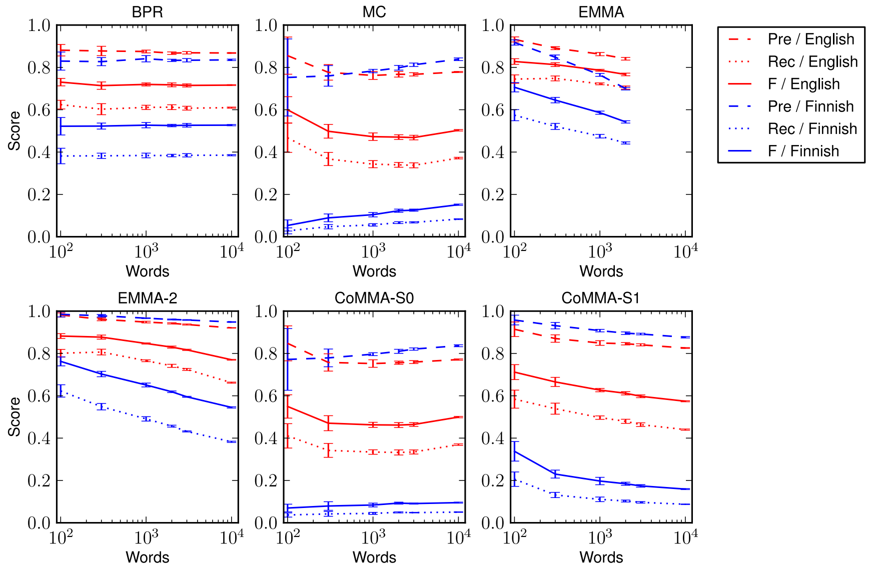

Figure 11. *The mean and standard deviation for precision, recall, and F-score of different evaluation methods with respect to the size of the evaluation data (100-10,000 words) for English and Finnish. The evaluated method is Morfessor Baseline.*

4.6. *Discussion*

As finding the β that gives the highest correlation of F_β to the application evaluations can be considered as tuning the evaluation metric, a relevant question is how general the found value is for given language and application. That is, if F_β optimized for the task is utilized for evaluating a new set of algorithms, will it actually give better correlations at all?

In order to test this, we used the β:s optimized for the IR correlations using the segmentation algorithms (Figure 6 on page 74) to calculate correlation using all the other (non-segmentation) algorithms. For English IR, there were 17 segmentation algorithms and 18 non-segmentation algorithms, and for Finnish IR, 15 for both. The obtained β:s and correlations are shown in Table 6. Using the β optimized for segmentation algorithms often gives higher correlations also for non-segmentation algorithms than the balanced F_1-score. In the cases that it does not, F_β and F_1 are either equal or very close. Moreover, in half of the cases, the correlation of F_β tuned for the non-segmentation algorithms (shown in last column) is only slightly (≤ 0.05) higher than the one tuned for the segmentation algorithms.

Table 6. *Correlations with F_β tuned for segmentation algorithms and tested on non-segmentation algorithms. For comparison, the fifth column shows the correlation of balanced F-score for non-segmentation algorithms and the last column shows the correlation of the F_β-score optimized using the non-segmentation algorithms. The higher of the correlations for F_1 and F_β is shown in bold.*

Language	*Method*	*Segmentation*		*Non-segmentation*		
		β	F_β	F_1	F_β	*best* F_β
English	EMMA	1.25	0.87	**0.73**	0.71	0.73
English	MC	0.6	0.62	0.43	**0.56**	0.70
English	CoMMA-B0	0.6	0.69	0.44	**0.60**	0.72
English	CoMMA-B1	0.9	0.66	0.69	**0.74**	0.76
English	EMMA-2	1.1	0.86	**0.74**	**0.74**	0.75
English	CoMMA-S1	1.0	0.70	**0.79**	**0.79**	0.84
English	CoMMA-S0	0.6	0.70	**0.67**	0.65	0.82
Finnish	EMMA	1.25	0.91	0.59	**0.61**	0.76
Finnish	MC	0.3	0.76	0.35	**0.73**	0.73
Finnish	CoMMA-B0	0.3	0.65	0.18	**0.47**	0.60
Finnish	CoMMA-B1	0.4	0.71	0.29	**0.52**	0.52
Finnish	EMMA-2	0.8	0.90	**0.55**	0.53	0.67
Finnish	CoMMA-S1	0.5	0.76	0.44	**0.62**	0.66
Finnish	CoMMA-S0	0.3	0.74	0.44	**0.61**	0.69

Note that in the experiment above, the set of algorithms for tuning the β and those testing it were quite different, as only the second set considered any non-concatenative processes at all. Thus, the β:s optimized for the whole set of algorithms evaluated so far are likely to provide good correlations also for novel sets of algorithms. However, this certainly does not mean that application evaluations are unnecessary for future evaluation campaigns. A simple reason is that the best unsupervised algorithms have actually outperformed the grammatically correct analyses for the applications evaluated in Morpho Challenges (Kurimo *et al.*, 2008; Kurimo *et al.*, 2009; Kurimo *et al.*, 2010c; Kurimo *et al.*, 2010b). In other words, the grammatically correct analysis is not likely to be the optimal solution for the applications.

The optimally correlated F_β of the soft isomorphic evaluations usually weights precision over recall, especially for agglutinative languages (Finnish and German). This indicates that undersegmentation is sometimes useful for the applications. While this may originate from some application-specific methods (developed usually for English), the phenomenon can also be considered, for example, in the context of the psycholinguistic discussion on whether inflected words are stored as full-forms or inferred from their morphological parts in human mind (see, e.g., Pinker and Ullman, 2002; Baayen, 2007).

Finally, we emphasize that the results are always dependent on the gold standard used for reference analyses. In particular, labels that have no direct correspondences in the surface forms pose a large problem for unsupervised approaches. Common examples are separate part-of-speech labels and null morphemes marking singular forms of nouns. Such labels are likely to encourage oversegmentation, especially with co-occurrence based metrics, as happened with the Arabic evaluations in Morpho Challenge 2009 (Kurimo *et al.*, 2010c). While the reference analyses used in this study are clean from most of such labels, some other peculiarities remain. For example, the shortened words in the English gold standard have the morphemes of the long form (e.g., *ad* has *advertise_V* and *ment_s*; *a-bomb* has *atom_N* and *bomb_N*).

4.7. *Summary*

We end this section by summarizing the experimental results for the isomorphic evaluation methods.

MC: the MC evaluation has only weak correlation to application evaluations with balanced F-score, but a decent one when precision is given more weight than recall. The evaluation, based on random sampling of morpheme-sharing word pairs, is designed for the case where there is a large number of analyzed words to compare. The naive treatment of alternative analyses makes the method vulnerable to gaming. Algorithms with low recall can be artificially boosted also by adding shared morphemes.

EMMA: EMMA gives high correlations to application evaluations even with balanced F-score. Even among F_β-scores, the correlations to the results of the IR task are the highest. EMMA is also robust to gaming both with ambiguity hijacking and morpheme padding. In addition, it provides the mapping from the predicted to the reference morphemes that can be used in qualitative evaluation. The main problems are the computational complexity and the memory requirements of the algorithm. The actual implementation uses integer linear programming, which could be replaced by the Hungarian algorithm, but the complexity will still be at least cubic with respect to the number of morphemes. Minor drawbacks are that the precision and recall do not have as practical interpretations as in soft isomorphic methods, and the measures results are overestimated for small data sets.

EMMA-2: like EMMA, EMMA-2 provides high correlations to application evaluations even with balanced F-score and is robust to gaming. Also similar to EMMA, it gives overestimated results with small data sets. Precision and recall of EMMA-2 behave similarly to those of the co-occurrence metrics and are more useful than in EMMA, but the matchings between predicted and reference morphemes are not as easy to use in qualitative evaluation. The main advantage over EMMA is that EMMA-2 is very quick to compute.

CoMMA: the S0 and S1 versions of CoMMA show a positive correlation to the IR results with balanced F-score, but the correlation is high only for English. With weighted F_β-score, the correlations vary from high (English) to moderate (German). B0 and B1 have clearly lower correlations in both cases. Excluding the isolated words from the evaluation has mixed effects on correlations: B0 has the lowest correlations of the variants, but S0 is among the best ones. More weighting is usually needed to get as good correlations to the application evaluations. However, it provides more stable results for small evaluation sets, for which the scores are otherwise overestimated. Similar to MC, the recall of the CoMMA evaluations is vulnerable to gaming with shared morpheme padding. However, they are more robust than MC with respect to ambiguity hijacking, especially CoMMA-S which uses a strict matching between the alternatives. All the variants have reasonable computation times up to 10,000 words.

5. Conclusions

Unsupervised learning of morphology has been an active research topic for over a decade, but there has not been any standard way of evaluating the algorithms based on linguistic reference analyses. While this is partly due to the lack of free and publicly available linguistic references, also the implementations and experimental comparisons of the evaluation methods have been missing. The situation has been improved by the yearly Morpho Challenge evaluations and the publication of the EMMA method (Spiegler and Monson, 2010). For this article, we have performed the most extensive meta-evaluation so far, using the large number of submissions to the Morpho Challenge competitions. Based on the experiments, we can give some recommendations for the usage of the evaluation methods.

While the emphasis of this work was on isomorphic evaluations, the results confirm that using boundary evaluation is sensible whenever it is applicable (i.e., both predicted analyses and reference analyses are segmentations). In addition to being robust, simple and intuitive, it provides high correlations to application evaluations. In the case that reference segmentations are not available, isomorphic evaluations provide reasonable correlations to the boundary evaluation. However, it should be kept in mind that the correlation depends both on the language and the reference analysis, which may require cleaning from, e.g., null morphemes.

Among the isomorphic evaluation methods, EMMA is recommended especially if either the goal is to get as close to the reference analysis as possible (one-to-one assignment provides detailed information) or a good correlation of the balanced F-score to the application evaluations is sought. However, computational complexity of the assignment prevents using it for large evaluation data sets. EMMA-2 maintains the strengths of EMMA, robustness and high correlation to application evaluations, while having substantially shorter computation times. The use of soft (many-to-one) assignment instead of the hard assignment of EMMA reduces the interpretability of the morpheme assignments, but increases the interpretability of precision and recall. The combination of robustness and efficiency makes it a strong candidate for any large-scale experiments and competitions.

CoMMA-S fixes the two main problems in the old MC evaluation. First, it removes the need of sampling and thus is more suitable to use with small evaluation sets. Second, it deals with alternative analyses in more robust manner. Compared to the assignment based methods, CoMMA-S loses in the strength of the correlations to application evaluations, in particular with balanced F-scores and morphologically rich languages. However, it can still be recommended for English (and possibly other mostly analytic languages), where it works as well as EMMA and EMMA-2 practically in all aspects. The advantage of CoMMA-S0 over EMMA or EMMA-2 is the stablility with respect to the size of the evaluation set, which helps comparing the results from a small development set to those of the final test set.

Implementations of the new evaluation methods, as well as the results for the individual algorithms submitted to Morpho Challenges, will be published at `http://research.ics.tkk.fi/events/morphochallenge/`.

Acknowledgements

This work was supported by the Academy of Finland in the project *Adaptive Informatics* and the Graduate School of Language Technology in Finland. We thank the anonymous reviewers for their valuable comments, and Mathias Creutz, Krista Lagus, and Christian Monson for useful discussions. The organization of Morpho Challenges have been supported by the 6th and 7th Framework Programmes of the European Commission under the PASCAL and PASCAL2 Networks of Excellence and the Cross-Linguistic Evaluation Forum (CLEF) of the PROMISE Network of Excellence. We are grateful to CLEF, University of Leipzig, University of Leeds, Computational Linguistics Group at University of Haifa, Stefan Bordag, Ebru Arisoy, Nizar Habash, Majdi Sawalha, Eric Atwell, and Mathias Creutz for making the data and gold standards in various languages available for Morpho Challenges. Finally, we warmly thank everyone who has participated in the Challenges. This work would not have been possible without their contributions.

6. References

Agirre E., Di Nunzio G. M., Ferro N., Mandl T., Peters C., “CLEF 2008: Ad Hoc Track Overview”, *Working Notes for the CLEF 2008 Workshop*, September, 2008.

Alkula R., “From Plain Character Strings to Meaningful Words: Producing Better Full Text Databases for Inflectional and Compounding Languages with Morphological Analysis Software”, *Information Retrieval*, vol. 4, p. 195-208, 2001. 10.1023/A:1011942104443.

Alpaydin E., *Introduction to Machine Learning*, The MIT Press, Cambridge, MA, USA, 2004.

Baayen R. H., “Storage and Computation in the Mental Lexicon”, *in* G. Jarema, G. Libben (eds), *The Mental Lexicon: Core Perspectives*, Elsevier, p. 81-104, 2007.

Baayen R. H., Piepenbrock R., Gulikers L., *The CELEX Lexical Database (CD-ROM)*, Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA, 1995. `http://wave.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC96L14`.

Baroni M., Matiasek J., Trost H., “Unsupervised Discovery of Morphologically Related Words Based on Orthographic and Semantic Similarity”, *Proceedings of the ACL-02 Workshop* *on Morphological and Phonological Learning*, Association for Computational Linguistics, Morristown, NJ, USA, p. 48-57, July, 2002.

Bernhard D., "Simple Morpheme Labelling in Unsupervised Morpheme Analysis", *Advances in Multilingual and Multimodal Information Retrieval, 8th Workshop of the Cross-Language Evaluation Forum, CLEF 2007, Budapest, Hungary, September 19-21, 2007, Revised Selected Papers*, vol. 5152 of *Lecture Notes in Computer Science*, Springer, p. 873-880, 2008.

Bernhard D., "MorphoNet: Exploring the Use of Community Structure for Unsupervised Morpheme Analysis", *Multilingual Information Access Evaluation I. Text Retrieval Experiments: 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfu, Greece, September 30 - October 2, 2009, Revised Selected Papers*, vol. 6241 of *Lecture Notes in Computer Science*, Springer, p. 598-608, 2010.

Bordag S., "Unsupervised and Knowledge-Free Morpheme Segmentation and Analysis", *Advances in Multilingual and Multimodal Information Retrieval, 8th Workshop of the Cross-Language Evaluation Forum, CLEF 2007, Budapest, Hungary, September 19-21, 2007, Revised Selected Papers*, vol. 5152 of *Lecture Notes in Computer Science*, Springer, p. 881-891, 2008.

Can B., Manandhar S., "Clustering Morphological Paradigms Using Syntactic Categories", *Multilingual Information Access Evaluation I. Text Retrieval Experiments: 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfu, Greece, September 30 - October 2, 2009, Revised Selected Papers*, vol. 6241 of *Lecture Notes in Computer Science*, Springer, p. 641-648, 2010.

Chan E., "Learning Probabilistic Paradigms for Morphology in a Latent Class Model", *Proceedings of the Eighth Meeting of the ACL Special Interest Group on Computational Phonology and Morphology at HLT-NAACL 2006*, Association for Computational Linguistics, New York City, USA, p. 69-78, June, 2006.

Chan E., *Structures and Distributions in Morphology Learning*, PhD thesis, Department of Computer and Information Science, University of Pennsylvania, 2008.

Christodoulopoulos C., Goldwater S., Steedman M., "Two Decades of Unsupervised POS Induction: How Far Have We Come?", *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Cambridge, MA, p. 575-584, October, 2010.

Creutz M., "Unsupervised Segmentation of Words Using Prior Distributions of Morph Length and Frequency", *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Sapporo, Japan, p. 280-287, July, 2003.

Creutz M., Lagus K., "Unsupervised Discovery of Morphemes", *Proceedings of the Workshop on Morphological and Phonological Learning of ACL-02*, p. 21-30, 2002.

Creutz M., Lagus K., "Inducing the Morphological Lexicon of a Natural Language From Unannotated Text", *Proceedings of the International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning (AKRR'05)*, Espoo, Finland, p. 106-113, 2005a.

Creutz M., Lagus K., "Unsupervised Morpheme Segmentation and Morphology Induction from Text Corpora Using Morfessor 1.0", Technical Report no. A81, Publications in Computer and Information Science, Helsinki University of Technology, 2005b.

Creutz M., Lagus K., “Unsupervised Models for Morpheme Segmentation and Morphology Learning”, *ACM Transactions on Speech and Language Processing*, January, 2007.

Creutz M., Lindén K., “Morpheme Segmentation Gold Standards for Finnish and English”, Technical Report no. A77, Publications in Computer and Information Science, Helsinki University of Technology, 2004.

Dasgupta S., Ng V., “High-Performance, Language-Independent Morphological Segmentation”, *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, Association for Computational Linguistics, Rochester, New York, p. 155-163, April, 2007.

Gao J., Johnson M., “A Comparison of Bayesian Estimators for Unsupervised Hidden Markov Model POS Taggers”, *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Honolulu, Hawaii, p. 344-352, October, 2008.

Goldsmith J., “Unsupervised Learning of the Morphology of a Natural Language”, *Computational Linguistics*, vol. 27, no. 2, p. 153-189, 2001.

Golénia B., Spiegler S., Flach P. A., “Unsupervised Morpheme Discovery with Ungrade”, *Multilingual Information Access Evaluation I. Text Retrieval Experiments: 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfu, Greece, September 30 - October 2, 2009, Revised Selected Papers*, vol. 6241 of *Lecture Notes in Computer Science*, Springer, p. 633-640, 2010a.

Golénia B., Spiegler S., Ray O., Flach P., “Morphological Analysis Using Morpheme Graph and Mixed-Integer Computation of Stable Models”, *in* M. Kurimo, S. Virpioja, V. T. Turunen (eds), *Proceedings of the Morpho Challenge 2010 Workshop*, Aalto University School of Science and Technology, Department of Information and Computer Science, Espoo, Finland, p. 25-29, September, 2010b. Technical Report TKK-ICS-R37. Extended abstract.

Hafer M. A., Weiss S. F., “Word Segmentation by Letter Successor Varieties”, *Information Storage and Retrieval*, vol. 10, no. 11-12, p. 371-385, 1974.

Hammarström H., Borin L., “Unsupervised Learning of Morphology”, *Computational Linguistics*, vol. 37, no. 2, p. 309-350, June, 2011.

Harris Z. S., “From Phoneme to Morpheme”, *Language*, vol. 31, no. 2, p. 190-222, 1955. Reprinted 1970 in *Papers in Structural and Transformational Linguistics*, Reidel Publishing Company, Dordrecht, Holland.

Hirsimäki T., Creutz M., Siivola V., Kurimo M., Virpioja S., Pylkkönen J., “Unlimited Vocabulary Speech Recognition with Morph Language Models Applied to Finnish”, *Computer, Speech and Language*, vol. 20, no. 4, p. 515-541, 2006.

Hirsimäki T., Pylkkönen J., Kurimo M., “Importance of High-Order N-Gram Models in Morph-Based Speech Recognition”, *IEEE Transactions on Audio, Speech and Language Processing*, vol. 17, no. 4, p. 724-732, May, 2009.

Jurafsky D., Martin J. H., *Speech and Language Processing*, 2nd edn, Prentice Hall, New Jersey, USA, 2008.

Karp R. M., “Reducibility Among Combinatorial Problems”, *in* R. E. Miller, J. W. Thatcher (eds), *Complexity of Computer Computations*, New York, Plenum, p. 85-103, 1972.

Keshava S., Pitler E., “A Simpler, Intuitive Approach to Morpheme Induction”, *Proceedings of the PASCAL Challenge Workshop on Unsupervised Segmentation of Words into Morphemes*, PASCAL European Network of Excellence, Venice, Italy, 2006.

Koehn P., “Europarl: A Parallel Corpus for Statistical Machine Translation”, *Proceedings of the 10th Machine Translation Summit*, Phuket, Thailand, p. 79-86, 2005.

Koehn P., Hoang H., Birch A., Callison-Burch C., Federico M., Bertoldi N., Cowan B., Shen W., Moran C., Zens R., Dyer C., Bojar O., Constantin A., Herbst E., “Moses: Open Source Toolkit for Statistical Machine Translation”, *Annual Meeting of ACL, demonstration session*, Czech Republic, June, 2007.

Kohonen O., Virpioja S., Klami M., “Allomorfessor: Towards Unsupervised Morpheme Analysis”, *Evaluating systems for Multilingual and Multimodal Information Access, 9th Workshop of the Cross-Language Evaluation Forum, CLEF 2008, Aarhus, Denmark, September 17-19, 2008, Revised Selected Papers*, vol. 5706 of *Lecture Notes in Computer Science*, Springer, 2009.

Kohonen O., Virpioja S., Lagus K., “Semi-Supervised Learning of Concatenative Morphology”, *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, Association for Computational Linguistics, Uppsala, Sweden, p. 78-86, July, 2010a.

Kohonen O., Virpioja S., Leppänen L., Lagus K., “Semi-Supervised Extensions to Morfessor Baseline”, *in* M. Kurimo, S. Virpioja, V. T. Turunen (eds), *Proceedings of the Morpho Challenge 2010 Workshop*, Aalto University School of Science and Technology, Department of Information and Computer Science, Espoo, Finland, p. 30-34, September, 2010b. Technical Report TKK-ICS-R37. Extended abstract.

Koskenniemi K., *Two-Level Morphology: A General Computational Model for Word-Form Recognition and Production*, PhD thesis, University of Helsinki, 1983.

Kuhn H. W., “The Hungarian Method for the Assignment Problem”, *Naval Research Logistics Quarterly*, vol. 2, p. 83-97, 1955.

Kumar S., Byrne W., “Minimum Bayes-Risk Decoding for Statistical Machine Translation”, *Proceedings of Human Language Technologies: The 2004 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, p. 169-176, 2004.

Kurimo M., Creutz M., Lagus K., “Unsupervised Segmentation of Words into Morphemes - Challenge 2005, An Introduction and Evaluation Report”, *Proceedings of the PASCAL Challenge Workshop on Unsupervised Segmentation of Words into Morphemes*, PASCAL European Network of Excellence, Venice, Italy, 2006.

Kurimo M., Creutz M., Turunen V., “Unsupervised Morpheme Analysis Evaluation by IR experiments – Morpho Challenge 2007”, *in* A. Nardi, C. Peters (eds), *Working Notes for the CLEF 2007 Workshop*, CLEF, September, 2007. Invited paper.

Kurimo M., Creutz M., Varjokallio M., “Morpho Challenge Evaluation Using a Linguistic Gold Standard”, *Advances in Multilingual and Multimodal Information Retrieval, 8th Workshop of the Cross-Language Evaluation Forum, CLEF 2007, Budapest, Hungary, September 19-21, 2007, Revised Selected Papers*, vol. 5152 of *Lecture Notes in Computer Science*, Springer, p. 864-873, 2008.

Kurimo M., Turunen V., Varjokallio M., “Overview of Morpho Challenge 2008”, *Evaluating Systems for Multilingual and Multimodal Information Access, 9th Workshop of the Cross-Language Evaluation Forum, CLEF 2008, Aarhus, Denmark, September 17-19, 2008, Revised Selected Papers*, vol. 5706 of *Lecture Notes in Computer Science*, Springer, 2009.

Kurimo M., Virpioja S., Turunen V., Lagus K., “Morpho Challenge 2005-2010: Evaluations and Results”, *Proceedings of the 11th Meeting of the ACL Special Interest Group on Compu-*

tational Morphology and Phonology, Association for Computational Linguistics, Uppsala, Sweden, p. 87-95, July, 2010a.

Kurimo M., Virpioja S., Turunen V. T., “Overview and Results of Morpho Challenge 2010”, *Proceedings of the Morpho Challenge 2010 Workshop*, Aalto University School of Science and Technology, Department of Information and Computer Science, Espoo, Finland, p. 7-24, September, 2010b. Technical Report TKK-ICS-R37.

Kurimo M., Virpioja S., Turunen V. T., Blackwood G. W., Byrne W., “Overview and Results of Morpho Challenge 2009”, *Multilingual Information Access Evaluation I. Text Retrieval Experiments: 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfu, Greece, September 30 - October 2, 2009, Revised Selected Papers*, vol. 6241 of *Lecture Notes in Computer Science*, Springer, p. 578-597, 2010c.

Lavallée J.-F., Langlais P., “Unsupervised Morphological Analysis by Formal Analogy”, *Multilingual Information Access Evaluation I. Text Retrieval Experiments: 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfu, Greece, September 30 - October 2, 2009, Revised Selected Papers*, vol. 6241 of *Lecture Notes in Computer Science*, Springer, p. 617-624, 2010.

Lignos C., “Learning from Unseen Data”, *in* M. Kurimo, S. Virpioja, V. T. Turunen (eds), *Proceedings of the Morpho Challenge 2010 Workshop*, Aalto University School of Science and Technology, Department of Information and Computer Science, Espoo, Finland, p. 35-38, September, 2010. Technical Report TKK-ICS-R37. Extended abstract.

Lignos C., Chan E., Marcus M. P., Yang C., “A Rule-Based Acquisition Model Adapted for Morphological Analysis”, *Multilingual Information Access Evaluation I. Text Retrieval Experiments: 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfu, Greece, September 30 - October 2, 2009, Revised Selected Papers*, vol. 6241 of *Lecture Notes in Computer Science*, Springer, p. 658-665, 2010a.

Lignos C., Chan E., Yang C., Marcus M. P., “Evidence for a Morphological Acquisition Model from Development Data”, *Proceedings of BUCLD 34*, Cascadilla Press, 2010b.

Lim H. S., Nam K., Hwang Y., “A Computational Model of Korean Mental Lexicon”, *in* O. Gervasi, M. Gavrilova, V. Kumar, A. Laganà, H. Lee, Y. Mun, D. Taniar, C. Tan (eds), *Computational Science and Its Applications — ICCSA 2005*, vol. 3480 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, p. 17-26, 2005.

Matthews P. H., *Morphology*, Cambridge Textbooks in Linguistics, 2nd edn, Cambridge University Press, 1991.

McNamee P., Mayfield J., “N-Gram Morphemes for Retrieval”, *Working Notes for the CLEF 2007 Workshop*, Budapest, Hungary, September, 2007.

Meila M., “Comparing Clusterings by the Variation of Information”, *in* B. Schölkopf, M. K. Warmuth (eds), *Learning Theory and Kernel Machines*, vol. 2777 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, p. 173-187, 2003.

Monson C., Carbonell J., Lavie A., Levin L., “ParaMor: Finding Paradigms Across Morphology”, *Advances in Multilingual and Multimodal Information Retrieval, 8th Workshop of the Cross-Language Evaluation Forum, CLEF 2007, Budapest, Hungary, September 19-21, 2007, Revised Selected Papers*, vol. 5152 of *Lecture Notes in Computer Science*, Springer, p. 900-907, 2008.

Monson C., Carbonell J., Lavie A., Levin L., “ParaMor and Morpho Challenge 2008”, *Evaluating Systems for Multilingual and Multimodal Information Access, 9th Workshop of* *the Cross-Language Evaluation Forum, CLEF 2008, Aarhus, Denmark, September 17-19, 2008, Revised Selected Papers*, vol. 5706 of *Lecture Notes in Computer Science*, Springer, p. 967-974, 2009.

Monson C., Hollingshead K., Roark B., “Simulating Morphological Analyzers with Stochastic Taggers for Confidence Estimation”, *Multilingual Information Access Evaluation I. Text Retrieval Experiments: 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfu, Greece, September 30 - October 2, 2009, Revised Selected Papers*, vol. 6241 of *Lecture Notes in Computer Science*, Springer, p. 649-657, 2010.

Munkres J., “Algorithms for the Assignment and Transportation Problems”, *Journal of the Society for Industrial and Applied Mathematics*, vol. 5, no. 1, p. 32-38, 1957.

Nicolas L., Farré J., Molinero M. A., “Unsupervised Learning of Concatenative Morphology Based on Frequency-Related Form Occurrence”, *in* M. Kurimo, S. Virpioja, V. T. Turunen (eds), *Proceedings of the Morpho Challenge 2010 Workshop*, Aalto University School of Science and Technology, Department of Information and Computer Science, Espoo, Finland, p. 39-43, September, 2010. Technical Report TKK-ICS-R37. Extended abstract.

Nießen S., Ney H., “Statistical Machine Translation With Scarce Resources Using Morpho-Syntactic Information”, *Computational Linguistics*, vol. 30, no. 2, p. 181-204, 2004.

Oflazer K., El-Kahlout İ. D., “Exploring Different Representational Units in English-to-Turkish Statistical Machine Translation”, *Proceedings of the Statistical Machine Translation Workshop at ACL 2007*, Association for Computational Linguistics, Prague, Czech Republic, p. 25-32, June, 2007.

Ogilvie P., Callan J., “Experiments Using the Lemur Toolkit”, *Proc. TREC ’01*, National Institute of Standards and Technology, special publication, Gaithersburg, Maryland, USA, p. 103-108, 2002.

Papineni K., Roukos S., Ward T., Zhu W.-J., “BLEU: A Method for Automatic Evaluation of Machine Translation”, *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL’02)*, Association for Computational Linguistics, Morristown, NJ, USA, p. 311-318, 2002.

Pinker S., Ullman M. T., “The Past and Future of the Past Tense”, *Trends in Cognitive Sciences*, vol. 6, p. 456-463, November, 2002.

Pirkola A., “Morphological Typology of Languages for IR”, *Journal of Documentation*, vol. 57, no. 3, p. 330-348, 2001.

Poon H., Cherry C., Toutanova K., “Unsupervised Morphological Segmentation with Log-Linear Models”, *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2009)*, Association for Computational Linguistics, p. 209-217, 2009.

Rosenberg A., Hirschberg J., “V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure”, *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, Association for Computational Linguistics, Prague, Czech Republic, p. 410-420, June, 2007.

Schone P., Jurafsky D., “Knowledge-Free Induction of Morphology Using Latent Semantic Analysis”, *Proceedings of the 2nd Workshop on Learning Language in Logic and the 4th Conference on Computational Natural Language Learning*, Association for Computational Linguistics, Morristown, NJ, USA, p. 67-72, 2000.

Schone P., Jurafsky D., “Knowledge-Free Induction of Inflectional Morphologies”, *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics*, Association for Computational Linguistics, Pittsburgh, USA, 2001.

Shalonova K., Golenia B., Flach P., “Towards Learning Morphology for Under-Resourced Fusional and Agglutinating Languages”, *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 5, p. 956-965, July, 2009.

Snover M. G., Jarosz G. E., Brent M. R., “Unsupervised Learning of Morphology Using a Novel Directed Search Algorithm: Taking the First Step”, *Proceedings of the ACL-02 Workshop on Morphological and Phonological Learning*, Association for Computational Linguistics, p. 11-20, July, 2002.

Snyder B., Barzilay R., “Unsupervised Multilingual Learning for Morphological Segmentation”, *Proceedings of ACL-08: HLT*, Association for Computational Linguistics, Columbus, Ohio, p. 737-745, June, 2008.

Spiegler S., *Machine Learning for the Analysis of Morphologically Complex Languages*, PhD thesis, Merchant Venturers School of Engineering, University of Bristol, April, 2011.

Spiegler S., Golénia B., Flach P. A., “DEAP: Deductive-Abductive Parsing for Morphological Analysis”, *in* M. Kurimo, S. Virpioja, V. T. Turunen (eds), *Proceedings of the Morpho Challenge 2010 Workshop*, Aalto University School of Science and Technology, Department of Information and Computer Science, Espoo, Finland, p. 44-48, September, 2010a. Technical Report TKK-ICS-R37. Extended abstract.

Spiegler S., Golénia B., Flach P. A., “Unsupervised Word Decomposition With the Promodes Algorithm”, *Multilingual Information Access Evaluation I. Text Retrieval Experiments: 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfu, Greece, September 30 - October 2, 2009, Revised Selected Papers*, vol. 6241 of *Lecture Notes in Computer Science*, Springer, p. 625-632, 2010b.

Spiegler S., Golénia B., Flach P. A., “Word Decomposition With the Promodes Algorithm Family Bootstrapped on a Small Labelled Dataset”, *in* M. Kurimo, S. Virpioja, V. T. Turunen (eds), *Proceedings of the Morpho Challenge 2010 Workshop*, Aalto University School of Science and Technology, Department of Information and Computer Science, Espoo, Finland, p. 49-52, September, 2010c. Technical Report TKK-ICS-R37. Extended abstract.

Spiegler S., Monson C., “EMMA: A Novel Evaluation Metric for Morphological Analysis”, *Proceedings of the 23rd International Conference on Computational Linguistics (COLING)*, August, 2010.

Tchoukalov T., Monson C., Roark B., “Morphological Analysis by Multiple Sequence Alignment”, *Multilingual Information Access Evaluation I. Text Retrieval Experiments: 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfu, Greece, September 30 - October 2, 2009, Revised Selected Papers*, vol. 6241 of *Lecture Notes in Computer Science*, Springer, p. 666-673, 2010.

Virpioja S., Kohonen O., Lagus K., “Unsupervised Morpheme Analysis with Allomorfessor”, *Multilingual Information Access Evaluation I. Text Retrieval Experiments: 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfu, Greece, September 30 - October 2, 2009, Revised Selected Papers*, vol. 6241 of *Lecture Notes in Computer Science*, Springer, p. 609-616, 2010.

Virpioja S., Lehtonen M., Hultén A., Salmelin R., Lagus K., “Predicting Reaction Times in Word Recognition by Unsupervised Learning of Morphology”, *Proceedings of Interna-*

tional Conference on Artificial Neural Networks (ICANN 2011), vol. 6791 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, Espoo, Finland, p. 275-282, 2011.

Virpioja S., Väyrynen J. J., Creutz M., Sadeniemi M., "Morphology-Aware Statistical Machine Translation Based on Morphs Induced in an Unsupervised Manner", *Proceedings of the Machine Translation Summit XI*, Copenhagen, Denmark, p. 491-498, September, 2007.

Yang M., Kirchhoff K., "Phrase-Based Backoff Models for Machine Translation of Highly Inflected Languages", *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006)*, Association for Computational Linguistics, Trento, Italy, p. 41-48, 2006.

Yarowsky D., Wicentowski R., "Minimally Supervised Morphological Analysis by Multimodal Alignment", *Proceedings of the 38th Meeting of the ACL*, Association for Computational Linguistics, p. 207-216, 2000.

Zeman D., "Unsupervised Acquiring of Morphological Paradigms from Tokenized Text", *Advances in Multilingual and Multimodal Information Retrieval, 8th Workshop of the Cross-Language Evaluation Forum, CLEF 2007, Budapest, Hungary, September 19-21, 2007, Revised Selected Papers*, vol. 5152 of *Lecture Notes in Computer Science*, Springer, p. 892-899, 2008.

Zeman D., "Using Unsupervised Paradigm Acquisition for Prefixes", *Evaluating Systems for Multilingual and Multimodal Information Access, 9th Workshop of the Cross-Language Evaluation Forum, CLEF 2008, Aarhus, Denmark, September 17-19, 2008, Revised Selected Papers*, vol. 5706 of *Lecture Notes in Computer Science*, Springer, p. 983-990, 2009.

Zhu X., "Semi-Supervised Learning Literature Survey", Technical Report no. 1530, Computer Sciences, University of Wisconsin-Madison, 2005.