Assisting non-expert speakers of under-resourced languages in assigning stems and inflectional paradigms to new word entries of morphological dictionaries*

Miquel Esplà-Gomis

Rafael C. Carrasco

Víctor M. Sánchez-Cartagena

Mikel L. Forcada

Felipe Sánchez-Martínez

Juan Antonio Pérez-Ortiz

Dep. de Llenguatges i Sistemes Informàtics, Universitat d'Alacant, E-03071 Alacant, Spain

{mespla,carrasco,vmsanchez,mlf,fsanchez,japerez}@dlsi.ua.es

Abstract

This paper presents a new method with which to assist individuals with no background in linguistics to create monolingual dictionaries such as those used by the morphological analysers of many natural language processing applications. The involvement of non-expert users is especially critical for under-resourced languages which either lack or cannot afford the recruitment of a skilled workforce. Adding a word to a morphological dictionary usually requires identifying its stem along with the inflection paradigm that can be used in order to generate all the word forms of the new entry. Our method works under the assumption that the average speakers of a language can successfully answer the polar question "*is x a valid form of the word w to be inserted?*", where *x* represents tentative alternative (inflected) forms of the new word *w*. The experiments show that with a small number of polar questions the correct stem and paradigm can be obtained from non-experts with high success rates. We study the impact of different heuristic and probabilistic approaches on the actual number of questions.

*This document is a postprint version of a paper published in 2016 in the journal *Language Resources and Evaluation*. The final publication is available at Springer via http://dx.doi.org/10.1007/s10579-016-9360-9. Use that link to get the full bibliographic reference of the article.

1 Introduction

Under-resourced languages that suffer from a lack of linguistic resources cannot usually afford the skilled labour required to create many of these resources. Under this assumption, methods that ease the involvement of a broader group of non-expert people can significantly reduce the development costs and speed up the creation of high-quality linguistic data for these under-resourced languages. In this work, we focus on the enlargement of monolingual morphological dictionaries, such as those used by the morphological analysers of many natural language processing applications. This kind of resource is usually created by trained professionals who master certain skills: an in-depth knowledge of the language or languages involved; advanced knowledge about linguistics and morphology; and expertise in the format and encoding used in the particular dictionary. Our proposal, however, is addressed towards average speakers who may not know the difference between, for example, an adjective and a noun, and who will not be required to learn any of the aspects relating to the encoding of the entries in the dictionary. The whole system works under the assumption that average speakers of a language can correctly answer the polar question "*is x a valid form of the word w to be inserted?*".

Among the many contexts in which morphological dictionary enlargement is needed, in this paper we focus on the scenario of non-expert *users* (in a broad sense) who want to introduce into the monolingual dictionaries of a rule-based machine translation (MT) system (Hutchins and Somers, 1992) the word forms found in an input text that are unknown to the system, so that it can subsequently correctly translate them;[1] note, however, that our method could be applied to the addition of entries to the morphological dictionaries used in many other natural language processing applications. When there is a source word form that the MT system is not able to analyse because it is not present in its source language (SL) morphological dictionary, our approach will help the user to insert the corresponding entry. If the user is bilingual, the method can also be used to insert the translation of the word into the target-language (TL) morphological dictionary and, once both monolingual entries have been inserted, the corresponding entry can be inserted in the bilingual dictionary without further human intervention.[2] Users are expected to be motivated to contribute to the task because they will notice an immediate improvement in the MT system merely by answering a few polar questions. As already stated, this minimal and simple interaction with the system and the involvement of regular speakers make our approach particularly suitable for rule-based MT systems involving under-resourced languages, whose development has traditionally been possible with the implication of non-expert volunteers with limited linguistic knowledge (Forcada et al, 2011); our approach consequently broadens the range of non-experts who may contribute to the creation of the required resources.

This approach could also be applied outside the scenario described previously. For

[1] It could also occur that the word form is not completely unknown, but it is not analysed the way in which it should because it is a homograph and has more than one possible morphological analysis.

[2] In rule-based MT, the SL morphological dictionary contains mappings between SL *word forms* (also called *surface forms*), that is, words as they are found in texts, and SL *lexical forms* (comprising lemma, part-of-speech and inflection information). The same applies to the TL dictionary. Bilingual dictionaries contain mappings between SL lexical forms and TL lexical forms.

instance, other non-expert individuals, who are not users of the MT system, can be recruited by means of crowdsourcing platforms (Wang et al, 2013) to collaboratively perform the task of inserting new entries into morphological dictionaries. Moreover, linguists themselves can also benefit from the approach because validating inflected forms of words may often be faster than choosing from among a list of paradigms.[3]

The objective of our work is to obtain a system which can be used not only to add the particular unknown word form (for example, *wants*) to the dictionary, but also to assist in discovering an appropriate *stem* and a suitable *inflection paradigm* so that all the word forms of the unknown word and their associated morphological inflection information (such as *wants, verb, present, 3rd person* or *wanting, verb, gerund*) can be additionally inserted in one go. The stem is the part of a word that is common to all its inflected forms. Inflection paradigms are commonly used in rule-based MT systems in order to group regularities in the inflection of a set of words;[4] for many languages,[5] a paradigm is usually defined as a collection of suffixes and their corresponding morphological information; e.g., the paradigm assigned to many common English verbs indicates that by adding the suffix *-ing* to the stem, the gerund is obtained; by adding the suffix *-ed*, the past is obtained; etc.[6] As formulated, the methodology described in this paper applies to suffix-inflecting languages; the method can be straightforwardly adapted to languages that inflect only by changing prefixes; in the case of languages with nonconcatenative (for example, template-based) morphologies (such as Arabic), inflection could also be written as context-sensitive rewriting operators grouped in paradigms, and the methodology could be similarly adapted.

In this work we assume that the paradigms for all possible words in the language are already included in the dictionary; the data used to test the methodology (see Section 6.1.1) actually shows a very early stabilisation of the number of paradigms.[7] In our experience with the Apertium free/open-source platform (Forcada et al, 2011), on which dictionary development is usually frequency-driven, stabilisation occurs quite early, as most "irregular" paradigms are associated with high-frequency words and from some point on, words are only added to highly-reused, "regular" paradigms. We shall also focus on monolingual dictionaries because insertion of information in the bilingual dictionaries of rule-based MT systems is usually straightforward, although in this case the correlation between the source and the target language entries in existing dictionaries can be further exploited to reduce the number of questions posed (Sánchez-Cartagena et al, 2012b).

The main novelty of this work when compared to our previous papers on this topic (Esplà-Gomis et al, 2011; Sánchez-Cartagena et al, 2012b; Esplà-Gomis et al, 2014) is that it exhaustively evaluates and compares the different models proposed in them. We also introduce a common framework with which to integrate all these ap-

[3] A Spanish morphological dictionary, for example, may contain around 500 inflection paradigms. Other morphologically rich languages may exceed this number.

[4] Paradigms ease dictionary management by reducing the quantity of information that needs to be stored, and by simplifying revision and validation because of the explicit encoding of regularities in the dictionary.

[5] Actually, for most European languages, Indo-European or not.

[6] Note that the concept of suffix here obviously refers to the phonological order, regardless of writing direction (left-to-right or right-to-left).

[7] Automatic acquisition of paradigms from monolingual corpora has already been explored (Monson, 2009), but this task is out of the scope of this work.

proaches. New experiments have additionally been carried out in order to clarify the impact of each model on the performance of the system. Moreover, our proposal has been evaluated for the first time with under-resourced languages such as Basque, Maltese and, to a lesser extent, Catalan.

The remainder of the paper is organised as follows. Section 2 discusses other works related to the proposal in this paper. The notation that will be used in the paper is described in Section 3. Section 4 describes and formalises the basics of our strategy to allow non-expert users to insert entries into morphological dictionaries; it is based on two critical components that strongly influence the number of polar questions that need to be posed to the user in order to properly insert the dictionary entry: a *feasibility score* and a *querying algorithm*. Section 5 starts by proposing a heuristic realisation of the two components, after which a more coherent and principled probabilistic reformulation of the feasibility score and querying algorithm based on binary decision trees and hidden Markov models is presented. Heuristic and probabilistic realisations of our method are then automatically evaluated and compared in Section 6, whereas a human evaluation that confirms that average speakers of a language can successfully answer our type of polar questions is described in Section 7. Finally, some concluding remarks are presented.

2 Related work

Multiple approaches with which to build MT systems with the help of non-expert users can be found in literature. Ambati et al (2010) propose asking non-expert bilingual informants to translate SL sentences in order to create a parallel corpus from which a statistical MT system (Koehn, 2010) is eventually built. Users interact through a crowdsourcing platform (Wang et al, 2013). An efficient active learning strategy (Haffari et al, 2009) is critical in this scenario: the SL sentences to be translated by the users should be those that, when included in the parallel corpus, cause the largest possible increase in the performance of the resulting statistical MT system.

Two of the most prominent works in literature in relation to the elicitation of native-speaker knowledge to build or improve rule-based MT systems are those by Font-Llitjós (2007) and McShane et al (2002). The former proposes a strategy with which to improve both transfer rules (used to cope with the grammatical divergences between the languages) and dictionaries by analysing the postediting process (that is, the process of correcting the output of the MT system) performed by a non-expert user through a dedicated interface. McShane et al (2002) design a framework with which to elicit linguistic knowledge from informants who are not trained linguists and use this information in order to build MT systems which translate into English; their framework provides users with extensive information about different linguistic phenomena to ease the elicitation task. Unlike the aforementioned approaches, our method is addressed towards pure rule-based MT systems in which a single translation is generated and no language model is used to rank a number of translation hypothesis; these kinds of systems are notably more sensitive to erroneous linguistic information. We also want to relieve users from the task of having to acquire linguistic skills.

Additional tools that ease the creation of linguistic resources for MT by users

with some linguistic background have also been developed. To this end, the *smart paradigms* devised by Détrez and Ranta (2012) help users to obtain the correct inflection paradigm for a new word to be inserted in a dictionary to be used for MT. A smart paradigm is a function that returns the most appropriate paradigm for a word given its lexical category, some of its word forms and some morphological inflection information. There are two important differences between this approach and ours: smart paradigms are created exclusively by human experts, and users of smart paradigms need to have some linguistic background; for instance, an expert could decide that to correctly choose the inflection paradigm of most verbs for a particular language the infinitive and the first person plural present indicative forms are needed; dictionary developers would then provide these two forms when inserting a new verb. Bartusková and Sedlácek (2002) also present a tool for the semi-automatic assignment of words to declension patterns; their system is based on a decision tree with a question in every node. Their proposal, unlike ours, works only for nouns and is aimed at experts because of the technical nature of the questions. Desai et al (2012) focus on paradigm assignment for verbs and use information collected from a corpus for each compatible paradigm; if the automatic method fails, users are then required to manually enter the correct paradigm.

As regards the automatic acquisition of morphological resources for MT, the work by Šnajder (2013) is of particular interest: he poses the choice of the most appropriate paradigm as a machine learning problem. Given the values of a set of features extracted from a monolingual corpus and from the orthographic properties of the lemmas, each compatible paradigm is classified as correct/incorrect by a *support vector machine* (Cristianini and Shawe-Taylor, 2000). The main difference between this approach and ours lies in the fact that their method works in a fully-automatic pipeline, while we use the models to minimise the number of questions posed to non-expert users.

The work by Kilbury et al (1992) shares objectives with this paper, but does so in a formalism that, unlike ours, explicitly takes into account syntactic (unification) features which are not available in our setting. More recently, Ahlberg et al (2014) have explored the task of constructing a semi-supervised system that accepts as input inflection tables containing all the possible word forms of a word, uses those tables to generalise a set of inflection paradigms similar to those proposed by Détrez and Ranta (2012), and subsequently allows the use of unannotated corpora to expand the inflection tables. In that work, the assignment of new words to paradigms follows heuristic approaches in order to compute confidence scores for each paradigm candidate in a way similar to that described in Section 5.1 of this paper. However, they use additional morphological information in order to better discriminate among candidates, whereas in our approach this information is not available and the system completely relies on the knowledge of non-expert users.

3 Notation

Let $P = \{p_i\}$ be the set of paradigms in a monolingual dictionary. Each paradigm p_i defines a set of pairs (f_{ij}, m_{ij}), where f_{ij} is a suffix[8] which is appended to stems to build new *word forms*, and m_{ij} is the corresponding morphological information. Given a *stem/paradigm* combination $c = t/p_i$ composed of a stem t and a paradigm p_i, the *expansion* $I(c)$ is the set of possible word forms resulting from appending each of the suffixes f_{ij} in p_i to t. For instance, an English dictionary may contain the stem *want-* assigned to a paradigm with suffixes $p_i = \{$*-ε*, *-s*, *-ed*, *-ing*$\}$ (hereafter the morphological information contained in p_i is omitted and only suffixes are shown), where ε denotes the empty string; the expansion $I(\text{want}/p_i)$ consists of the set of word forms *want*, *wants*, *wanted* and *wanting*.

Given a new word form w to be added to a monolingual dictionary, the objective is to find both the stem $t \in \Pr(w)$, where $\Pr(w)$ is the set of all possible prefixes of w including ε and w itself, and the paradigm p_i such that $I(\text{w}/p_i)$ is the set of word forms which contains all the correct forms of the unknown word.

4 General method for knowledge elicitation

This section formalises our approach to the process that needs to be carried out in order to select the most appropriate polar questions to be posed to the user in order to elicit the corresponding monolingual dictionary entry from the answers provided.

4.1 Paradigm selection

In order to find the correct stem/paradigm combination for a new word form w, the first step is to obtain the set L that contains all the stem/paradigm combinations c_n compatible with w.[9] When word forms in the language of the dictionary are created by adding suffixes to the stem, this can be efficiently determined by using a *generalised suffix tree* (McCreight, 1976) containing all the possible suffixes included in the paradigms in P. As an example, consider a simple dictionary with only four paradigms: $p_1 = \{$*-ε*, *-s*$\}$; $p_2 = \{$*-y*, *-ies*$\}$; $p_3 = \{$*-y*, *-ies*, *-ied*, *-ying*$\}$; and $p_4 = \{$*-a*, *-um*$\}$. If the new word form to be inserted into the dictionary is *w=policies*, the set with the compatible stem/paradigm combinations is $L = \{c_1$*=policies*$/p_1$, c_2*=policie*$/p_1$, c_3*=polic*$/p_2$, c_4*=polic*$/p_3\}$.

4.2 Querying the user with polar questions

Once the set L of compatible stem/paradigms has been obtained, the user must decide which of the elements in L is the correct one for the word w, that is, which is the

[8] As discussed in the introduction, although our approach has been evaluated with languages that generate word forms by adding suffixes to stems (most European languages), it could straightforwardly be adapted to languages that inflect by changing prefixes, and with a little more effort to languages that show nonconcatenative inflection.

[9] We consider that a stem/paradigm combination c_n is compatible with a word form w if the stem of c_n concatenated to one of the suffixes in the paradigm results in the surface form w.

candidate $c_n \in L$ whose expansion $I(c_n)$ contains exactly all the valid forms of w. To that end, the user is iteratively asked to confirm whether a word form w' from $I(c_n)$ is a valid form of w. After each question, the candidates from L that are not compatible with the answer are discarded; the process continues until there is a single stem/paradigm pair in L, which will be that assigned to the new entry in the dictionary. The process comprises the following steps:

1. *Paradigm scoring.* A *feasibility score* is computed for each compatible stem/paradigm $c_n \in L$ using a large monolingual corpus C. Following the previous example, the word forms for the different candidates would be: $I(c_1)$=$\{$*policies*, *policiess*$\}$; $I(c_2)$=$\{$*policie*, *policies*$\}$; $I(c_3)$=$\{$*policy*, *policies*$\}$; and $I(c_4)$=$\{$*policy*, *policies*, *policied*, *policying*$\}$. A large English monolingual corpus would probably contain evidence that suggests that c_3 is the most likely candidate, and it would probably obtain the highest feasibility score. Different systems can be used to score paradigms: a heuristic approach that simply accounts for the number of word forms found in the monolingual corpus is presented in Section 5.1.1, after which a more sophisticated approach based on hidden Markov models is presented in Section 5.2.1.

2. *Selection of word forms (querying algorithm).* The best candidate is chosen from L by querying the user about whether or not some word forms w' for some of the compatible stem/paradigms $c_n \in L$ are correct forms of w. The queries are polar questions with the possible answers *yes* and *no*. The following actions are carried out depending on the user's answer to the query about w':

   - if it is accepted, all $c_n \in L$ for which $w' \notin I(c_n)$ are removed from L;
   - if it is rejected, all $c_n \in L$ for which $w' \in I(c_n)$ are removed from L.

   This process is repeated until $|L| = 1$. The remaining candidate in L will then be used to insert a new entry into the dictionary. The criterion followed to choose, in each step, the word form w' that the users will be asked (from here on, the *querying algorithm*) should ensure that as few questions as possible are posed to the user before obtaining the solution.

   The querying algorithm uses the feasibility score described above and the membership relation between the different word forms and the candidate stem/paradigm pairs as sources of information to reduce the number of questions. As with the evaluation of the paradigm scoring strategy, a heuristic querying algorithm is presented in Section 5.1.2, and a refinement based on decision trees is later described in Section 5.2.2.

When more than one paradigm provides exactly the same set of suffixes but with different part of speech or inflection information, no additional polar question can be asked in order to discriminate between them and the iterative querying process consequently finishes with $|L| > 1$. For example, in Spanish many adjectives such as *alto* and nouns such as *gato* are inflected identically. Two paradigms that produce the same collection of suffixes $\{$*-o* (masculine, singular), *-a* (feminine, singular), *-os* (masculine,

plural), *-as* (feminine, plural)} but with different morphological information are therefore defined in the monolingual dictionary (the stem *alt-* is assigned to the inflection paradigm whose part of speech is *adjective*, while *gat-* is assigned to the paradigm whose part of speech is *noun*). This issue also affects paradigms with the same part of speech: *abeja* and *abismo* are nouns that are inflected identically; *abeja* is however feminine, whereas *abismo* is masculine. When adding unknown words such as *gato* or *abeja*, no polar question can consequently be asked in order to discriminate between both paradigms. The solution to this issue is out of the scope of this paper.[10] In our experiments, when all the candidate stem/paradigm pairs in L generate the same set of word forms, if one of the candidates in L is the stem/paradigm pair used as a reference, the result of inserting the word is considered successful.

5 Heuristic and probabilistic realisations of the general method

Having described the general strategy with which to allow non-expert users to insert entries into morphological dictionaries, in this section two realisations of the general method described in the previous section are presented. Firstly, in Section 5.1 an approach based on a set of intuitive heuristics is discussed. After that, in Section 5.2 a more coherent and principled probabilistic reformulation of the feasibility score and querying algorithm based on binary decision trees and hidden Markov models (HMMs) is presented. Both alternatives will be evaluated and compared in Section 6.

5.1 Heuristic approach

5.1.1 Heuristic feasibility score

A feasibility score is assigned to each stem/paradigm candidate $c_n \in L$ using a large monolingual corpus C. This score should be higher for the candidates c_n that are more likely to be the correct one, according to the evidence found in the monolingual corpus. The more accurate the feasibility score, the fewer the questions that will be posed by the querying algorithm.

The heuristic feasibility score used in the experiments presented in this section has been defined under the assumption that the more word forms in $I(c_n)$ that are found in the monolingual corpus C, the more likely it is that the stem/paradigm candidate c_n is the most appropriate one. One possible way to compute the score is therefore

$$\text{Score}(c_n) = \frac{\sum_{\forall w' \in I(c_n)} \text{Appear}_C(w')}{|I(c_n)|^{\phi}},$$

where $\text{Appear}_C(w')$ is a function that returns 1 when the inflected form w' appears in the corpus C and 0 otherwise, and $I(\cdot)$ is the expansion function as defined previously. The exponent $\phi \leq 1$ is used to avoid very low scores for large paradigms which include

[10] Sánchez-Cartagena et al (2012a) propose a model based on an n-gram language model of lexical categories and morphological inflection information to perform the disambiguation.

a large number of suffixes. Preliminary experiments were carried out in order to find the most adequate value of ϕ. A set of words in the Spanish monolingual dictionary of the Spanish–Catalan[11] language pair in the Apertium MT platform (Forcada et al, 2011) was chosen and inserted following the strategy described in Section 4. We found that the values that maximised the number of words of the test set for which the best paradigm was scored with the highest feasibility score were between 0.4 and 0.6. A value of $\phi = 0.5$ was consequently used in the experiments reported in this paper.

To continue with the example shown in Section 4, in which the expansions of candidate stem/paradigm pairs are $I(c_1)$={*policies*, *policiess*}, $I(c_2)$={*policie*, *policies*}, $I(c_3)$={*policy*, *policies*}, and $I(c_4)$={*policy*, *policies*, *policied*, *policying*}, using a large monolingual English corpus C, the word forms *policies* and *policy* will be easily found; the other forms (*policie*, *policiess*, *policied* and *policying*) will not be found. The resulting scores could be, for example: Score(c_1)=0.71, Score(c_2)=0.71, Score(c_3)=1.41, Score(c_4)=1.

One potential problem with the previous formula is that all the inflections in $I(c_n)$ are taken into account, including those that, although morphologically correct, are not very usual in the corpus. To overcome this, Score(c_n) is redefined as

$$\text{Score}(c_n) = \frac{\sum_{\forall w' \in I'_C(c_n)} \text{Appear}_\text{C}(w')}{|I'_C(c_n)|^\phi},$$

where $I'_C(c_n)$ is the difference set

$$I'_C(c_n) = I(c_n) \setminus \text{Unusual}_C(c_n).$$

The function $\text{Unusual}_C(c_n)$ uses the words in the dictionary already assigned to p_i as a reference to obtain those of the inflections generated by p_i that are not usual in the corpus C. Let $T(p_i)$ be a function that retrieves the set of stems in the dictionary assigned to the paradigm p_i. For each of the suffixes f_{ij} of the corresponding paradigm our system computes

$$\text{Ratio}(f_{ij}, p_i) = \frac{\sum_{\forall t \in T(p_i)} \text{Appear}_\text{C}(t \cdot f_{ij})}{|T(p_i)|},$$

where "$\cdot$" is the string concatenation operator, and builds the set $\text{Unusual}_C(c_n)$ by concatenating the stem t to all the suffixes f_{ij} with $\text{Ratio}(f_{ij}, p_i)$ under a given threshold Θ.

5.1.2 Heuristic querying algorithm

The querying algorithm chooses, in each step of the iterative querying process described in Section 4.2, the word form that the user will have to validate. The algorithm should select the word forms in such a way that a single paradigm is obtained with as few questions as possible. To that end, the algorithm can use the feasibility score in

[11] Note that the language pair is indicated here because Apertium has slightly different monolingual dictionaries depending on the particular MT system in which they are used. As expected, the evaluation only uses the monolingual dictionary.

order to predict the user's answer, and the membership relation between the different word forms and the candidate stem/paradigm pairs in order to reduce the size of the set of candidates L as fast as possible.

The heuristic querying algorithm treats L as a list and sorts it in descending order by $\text{Score}(c_n)$. The membership relation is defined by the function $G(w', L)$, which returns the number of candidates $c_n \in L$ for which $w' \in I(c_n)$. Depending on the elements contained in L, the criterion followed by the algorithm in order to select the word form to be queried can be either *confirmation* or *discarding*. The algorithm starts in confirmation mode.

Confirmation mode. In this mode, it is assumed that all the word forms generated by the best candidate with highest score $c_1 \in L$ are correct. Consequently, the user is asked about the inflection $w' \in I(c_n)$ with the lowest value for $G(w', L)$ because, if it is accepted, a significant part of the paradigms in L will be removed from it. The algorithm keeps working in confirmation mode until:

- only one single candidate remains in L (it will be used as the final stem/paradigm pair inserted into the dictionary); or
- all the word forms $w' \in I(c_1)$ are generated by all the remaining candidates in L. In this situation, if a word form $w' \in I(c_1)$ were accepted by the user, the list L would remain unchanged. If it were discarded, L would become empty. The algorithm moves to discarding mode in order to break this lockout.

Discarding mode. In this mode, the system has accepted c_1 as a possible solution, but it needs to check whether any of the remaining candidates in L is more suitable. The new strategy is therefore to ask the user about those inflections $w' \notin I(c_1)$ with the highest possible value for $G(w', L)$. This process is repeated until

- only c_1 remains in L (it will be used as the final stem/paradigm pair inserted into the dictionary); or
- an inflection $w' \notin I(c_1)$ is accepted. This means that some of the other candidates are better than c_1.

If the second situation holds, the system removes c_1 from L and goes back to the confirmation mode. In both modes, if there are multiple word forms with the same value for $G(w', L)$, the system chooses the one with the highest $\text{Ratio}(f_{ij}, p_i)$, that is, the most usual in C and, consequently, the most likely to be familiar to the user.

5.2 Probabilistic approach

This section presents more rigorous and principled alternatives for the feasibility score and the querying algorithm, which are, in principle, meant to replace the intuitive heuristics defined in Section 5.1. In particular, hidden Markov models (Rabiner, 1989) are used to compute the feasibility score (Section 5.2.1), while binary decision trees are the foundation of the new querying algorithm (see Section 5.2.2).

5.2.1 Paradigm scoring with hidden Markov models

The heuristic approach used to compute the feasibility score of a candidate stem/paradigm pair described in Section 5.1.1 is based on the fraction of word forms found in a monolingual corpus. This kind of score can, however, be misleading under certain circumstances:

- when the word forms generated by the correct paradigm cannot be found at all in the corpus;
- even when they are found in the monolingual corpus, they may not represent a correct form of the word to be inserted into the dictionary. Consider, for instance, that the homograph word *complete*, that has been found in the sentence *I needed complete silence*, is to be inserted into the dictionary. Clearly, a paradigm that generates, among others, the word forms *completed* and *completing* (their attached morphological information would state that their lexical category is verb) should not obtain a high feasibility score. However, the heuristic scoring method described in Section 5.1.1 would assign a high feasibility score to that candidate paradigm if most of the word forms generated can be found in a corpus.

These two limitations of the heuristic feasibility score can be addressed by considering the sentence (context) in which the word to be inserted is found. To that end, a solution based on first-order hidden Markov models (HMM) is proposed. With HMMs, homography is partially dealt with as words are not considered in isolation and context is taken into account: the HMM determines the most likely part of speech and therefore prioritises the appropriate paradigms. An HMM model is a system that emits a sequence of observable outputs. Each time that it emits an output, its internal state can change, and the probability of emitting each observable output depends on the state, which cannot be observed (it is *hidden*). More formally, a first-order HMM is defined as $\lambda = (\Gamma, \Sigma, A, B, \pi)$, where Γ is the set of states, Σ is the set of observable outputs, A is the $|\Gamma| \times |\Gamma|$ matrix of state-to-state transition probabilities, B is the $|\Gamma| \times |\Sigma|$ matrix with the probability of each observable output $\sigma \in \Sigma$ being emitted from each state $\gamma \in \Gamma$, and the vector π, with dimensionality $|\Gamma|$, defines the initial probability of each state.

With this approach, the set of HMM states matches the set of all the paradigms in the dictionary, and the set of HMM observable outputs is obtained as the union of the suffixes produced by all these paradigms, that is, $\Sigma = \bigcup_{p_i \in P} \bigcup_{(f_{ij}, m_{ij}) \in p_i} f_{ij}$. The HMM for paradigm scoring is trained analogously to how HMMs are trained for unsupervised part-of-speech tagging (Cutting et al, 1992), that is, by using the Baum-Welch algorithm (Baum, 1972) with an untagged corpus. The training corpus is built from a monolingual corpus C as described below (Table 1 depicts an example sentence and the training data extracted from it):

- the entries in the monolingual dictionary in which the new words will be inserted are expanded in order to obtain the set F of all possible word forms. Let D be the set of entries present in the aforementioned dictionary; the set of all possible word forms is computed as $F = \bigcup_{t/p_i \in D} I(t/p_i)$;
- each word form w in the corpus that belongs to F is labelled with the corresponding paradigm (state) according to the dictionary. Since a word may be a

Table 1: Training data extracted from an example sentence, assuming that the dictionary only contains the paradigms $p_1 = \{$*-ε*, *-s*$\}$; $p_2 = \{$*-y*, *-ies*$\}$; $p_3 = \{$*-y*, *-ies*, *-ied*, *-ying*$\}$; $p_4 = \{$*-a*, *-um*$\}$; and $p_5 = \{$*-ε*$\}$, the word *today* is not in the dictionary, and the remaining words are.

Sentence (word forms):	the	baby	is	crying	today
Observable output:	*-ε*	*-y*	*-ε*	*-ying*	*-y*
States:	p_5	p_2	p_5	p_2	p_1
					p_2
					p_3
					p_5

homograph, if more than one paradigm generates w, the word is labelled with as many paradigms as found in the monolingual dictionary. With regard to the observable output assigned to w, since the HMM training architecture only allows a single observable output per element, its value is the longest suffix of w that can be found in the set of HMM observable outputs;

- each word form w' that cannot be found in F is labelled with the set of paradigms obtained from the set of its compatible candidates, as described in Section 4. The observable output assigned to w is again the longest suffix of w' that can be found in the set of HMM observable outputs.

Note again that, as mentioned in the introduction, for prefix-inflecting languages the observables could be prefixes and the method could be straightforwardly applied. In the case of other nonconcatenative inflection schemes such as the templatic morphology found in Arabic, observables could be context-sensitive rewriting operators. The main obstacle would be defining these HMM observables in a way that they can be easily extracted from unknown words, but the basics of the methodology could be adapted in a similar manner.

Once the HMM has been trained, the feasibility score of the different candidate paradigms is computed with the help of the input sentence in which the word to be inserted into the monolingual dictionary is present. Words in the input sentence are analysed as has been done with the sentences in the training corpus. Assuming that the word to be inserted is in position t of the input sentence, the feasibility score $Score(c_n)$ for each candidate $c_n \in L$ is computed by applying the following equation, which corresponds to Eq. (27) in the tutorial by Rabiner (1989):

$$Score(c_n) = q_t(c_n) = \frac{\alpha_t(c_n)\beta_t(c_n)}{\sum_{m=1}^{|L|} \alpha_t(c_m)\beta_t(c_m)} \quad (1)$$

This equation accounts for the product of the total probability of all sequences of states that go through state c_n at position t normalised by the sum of probabilities of all state sequences, that is, considering all the compatible candidates for the word form at position t. Given state c_n at position t, $\alpha_t(c_n)$ accounts for the forward probability

of the sub-sentence from the beginning of the sentence to position t, whereas $\beta_t(c_n)$ corresponds to the backward probability of the sub-sentence from position $t+1$ to the end of the sentence (Rabiner, 1989).

5.2.2 Selecting the terms to be queried about with binary decision trees

As in Section 5.1, the heuristic querying algorithm assumes that the user will accept any word form generated from the candidate stem/paradigm with the highest score, and selects the word form whose acceptance causes the discarding of the highest number of candidates. However, that algorithm has a number of limitations that are addressed by the new querying algorithm presented in this section:

- it is not able to detect when a higher number of candidates can be discarded if a word form is rejected by the user, and consequently it does not query the user about that word form even when its feasibility score is low (the score thus suggests that the word is likely to be rejected);
- if the feasibility score is not sufficiently reliable, i.e., the user does not accept word forms from the candidate paradigm with the highest feasibility score, the number of questions posed can be incremented dramatically.

In other words, a better querying algorithm should be more robust to incorrect feasibility scores and balance better the number of candidate stem/paradigms discarded when a word form is accepted, the number of candidate stem/paradigms discarded when a word form is rejected, and the likelihood of a word of being accepted or rejected according to the feasibility score. This behaviour is achieved with the use of binary decision trees built with the ID3 *Iterative Dichotomizer 3* algorithm (Quinlan, 1986). This algorithm follows a greedy approach to build each tree and the resulting trees are therefore sub-optimal. In each iteration, it selects the most appropriate attribute to split the data set. The algorithm starts from the root of the tree with the whole data set S. In each iteration, the attribute a that provides the highest information gain is picked to split S. A child node is then created for each possible value of a, with a new data set that contains only the elements that match that value. The information gain measures the difference in entropy before and after S is split. The entropy of a data set S is computed as

$$H(S) = -\sum_{x \in X} p(x) \log_2 p(x), \tag{2}$$

where X is the set of classes and the probability $p(x)$ of class x is usually computed as the proportion of elements of S that belong to the class x. The information gain $\mathrm{IG}(a,S)$ obtained when the data set is split by an attribute a is obtained as

$$\mathrm{IG}(a,S) = H(S) - \sum_{t \in T} p(t) H(t), \tag{3}$$

where T is the set of subsets obtained as a result of splitting S with the attribute a, $p(t)$ is usually calculated as the proportion of data points in t to the proportion of data points in S, and $H(t)$ is the entropy of the subset t.

Table 2: Example of the candidate stem/paradigm pairs and feasibility scores obtained when attempting to insert the word from *copies* into an English monolingual dictionary and following the heuristic approach presented in Section 5.1.1 in order to compute the feasibility scores. The remainder of the process is described in Section 5.2.2

c_n	$I(c_n)$	Feasibility score
$c_1 = (\text{copie}, p_1)$	{copie, copies}	0.31
$c_2 = (\text{cop}, p_2)$	{copy, copies}	0.25
$c_3 = (\text{copies}, p_1)$	{copies, copiess}	0.23
$c_4 = (\text{cop}, p_3)$	{copy, copies, copied, copying}	0.21

A decision tree can be used to implement a more robust querying algorithm. For each word form w to be inserted into the monolingual dictionary, the corresponding decision tree is built by means of the ID3 algorithm as follows:

- The data set S is built from all the stem/paradigm pairs compatible with w.
- The class of each data point is the corresponding stem/paradigm pair.
- The feature set is made up of the set of different word forms, that is $\cup_{c_n \in L} I(c_n)$.
- There are only two possible feature values: *yes* and *no*. The resulting decision tree is therefore binary.

The tree is traversed only once: the values of the features of the data point to be classified are the answers provided by the user. Note that if the proportion of data points that belong to class x were used to compute $p(x)$ in Equation 2, all the candidate stem/paradigm pairs would obtain the same probability (the data set from which the decision tree is built contains a single instance of each stem/paradigm pair). However, we can take advantage of the feasibility score computed by means of the HMMs described previously and assign that value to $p(x)$. Similarly, $p(t)$ in Equation 3 is calculated as the sum of the feasibility scores of the paradigms in t divided by the sum of the feasibilities of the paradigms in S. It has been empirically observed in the experiments presented in Section 6 that, when compared to a decision tree in which the usual definitions of $p(x)$ and $p(t)$ are used, these new definitions of $p(x)$ and $p(t)$, which take into account the feasibility score, reduce the depth of the leaf nodes that represent the candidate stem/paradigm pairs with highest feasibility scores only when the difference between feasibility scores is relatively high. This occurs at the expense of increasing the depth of the leaf nodes that represent the candidate stem/paradigm pairs with lower feasibility scores.

Let us illustrate the process with an example. Consider the paradigms $p_1 = \{$*-ε*, *-s*$\}$; $p_2 = \{$*-y*, *-ies*$\}$; $p_3 = \{$*-y*, *-ies*, *-ied*, *-ying*$\}$. If the word form *copies* (from the verb *copy*) were to be inserted into the monolingual dictionary, the candidate stem/paradigm pairs described in Table 2 would be obtained. Let us also suppose that the monolingual corpus used is not very reliable and the feasibility scores described in the table are obtained. The heuristic querying algorithm would need 3 queries to obtain the final

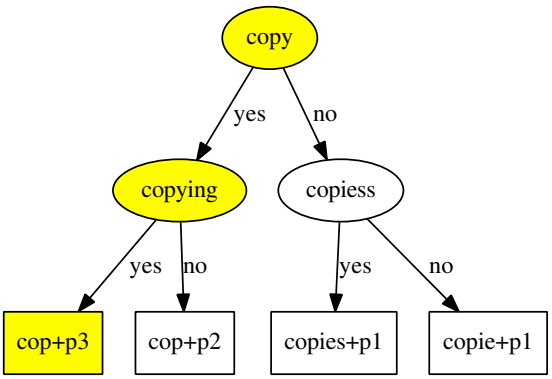

Figure 1: Binary decision tree generated by applying the ID3 algorithm to the candidate stem/paradigm pairs listed in Table 2.

stem/paradigm. First, it would ask the user to validate *copie*, the word form from the highest scored paradigm that causes the discarding of the highest amount of candidates when it is accepted. However, it would be rejected by the user, and only the candidate c_1 would be discarded. The heuristic querying algorithm would then choose *copy* by following the same criterion. It would be accepted and the two remaining candidates would be c_2 and c_4. Finally, since all the word forms generated by the paradigm with the highest feasibility score (c_2) are also generated by the remaining paradigms (c_4), the algorithm enters discarding mode and chooses *copying*.

If the querying algorithm based on decision trees were used, the binary decision tree depicted in Figure 1 would be obtained; note that only 2 questions are needed in order to attain the correct paradigm (the path is highlighted). Given that the feasibility score of the different candidates is similar, all the leaf nodes have the same depth.[12]

Let us now assume that the feasibility scores are more accurate, and the correct candidate, c_4, receives a very high feasibility score. The resulting decision tree, together with the new feasibility scores, is depicted in Figure 2. In this case, a single question is needed in order to attain the correct stem/paradigm pair. The heuristic querying algorithm would also need a single question. This example shows that the querying algorithm based on binary decision trees is more efficient than the heuristic algorithm when the feasibility score is not accurate, but it is also simultaneously able to take advantage of an accurate feasibility score as the heuristic algorithm does. This fact is confirmed by the experiments presented in the next section.

In summary, the use of a binary decision tree built with the ID3 algorithm allows the limitations of the heuristic querying algorithm to be overcome. The binary decision tree considers both the user's affirmative and negative answers and takes into account the feasibility scores thanks to the proposed modification of $p(x)$ and $p(t)$. In addition, it is more robust to unreliable feasibility scores, as illustrated in the previous example.

[12]This property is not guaranteed by the ID3 algorithm and it depends on the particular membership relation between the different word forms and the candidate stem/paradigm pairs. The tree becomes less *balanced* as the differences between the feasibility scores of the different candidates grow.

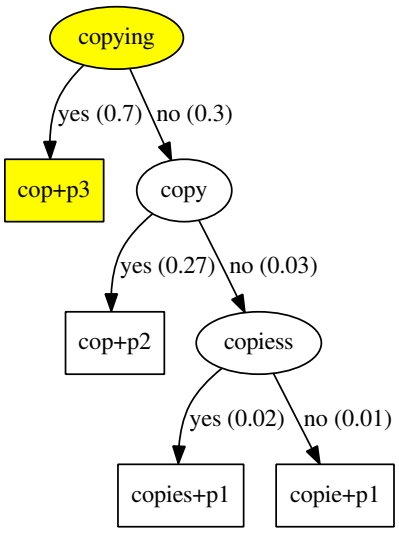

Figure 2: Binary decision tree generated by applying the ID3 algorithm to the candidate stem/paradigm pairs listed in Table 2, assuming that the probabilistic feasibility scores of each candidate stem/paradigm pair are those annotated in the tree itself.

Note that when using decision trees there are not two different modes of operation as with the heuristic approach (see Section 5.1.2).

6 Experiments and results

This section describes the automatic evaluation developed in order to compare the heuristic approach presented in Section 5.1 and the probabilistic approach presented in Section 5.2. In this experimental set-up, non-expert users, towards whom this method is eventually addressed, are replaced with an oracle that always chooses the answer that leads to the correct paradigm. Interferences caused by potential human errors are thus avoided. Note that the viability of the framework presented in this paper in a real-life scenario will be proved in the experiments described in Section 7 in which human evaluators took part.

6.1 Experimental set-up

The automatic evaluation consisted of simulating the addition of a set of words to the monolingual dictionary of selected language pairs of the Apertium MT platform (Forcada et al, 2011). In particular, four languages were involved in this experiment: Catalan, Basque, Maltese and Spanish, which were chosen from among a collection of reasonably mature MT systems in the Apertium repository. Spanish was included in the evaluation in order to have a reference for the performance of our approach with a well-resourced language, while Catalan, Basque, and Maltese were chosen since they

are under-resourced languages that belong to different language families. Our criterion on which languages can be considered under-resourced is based on the study driven by the META-NET network (Rehm and Uszkoreit, 2013).[13] Table 7 in that study compares 30 European languages as regards the amount of resources and different technologies, such as machine translation or text analysis, available for them. According to this study, only Spanish attains a moderate coverage of resources, while Catalan and Basque have a fragmentary coverage and Maltese has a weak coverage.

The monolingual dictionaries used in our experiments correspond to the Spanish–Catalan,[14] Spanish–Basque,[15] Arabic–Maltese,[16] and Spanish–English[17] MT systems, respectively. Six different test sets were built for the Spanish dictionary, while three test sets were built for the remaining languages: each of these sets contained a set of word forms to be inserted into each dictionary, and a context sentence in the corresponding language for each word form. The average number of questions needed to obtain the correct paradigm was computed for the following three systems:

- the heuristic approaches for the paradigm scoring and the querying algorithm described in Section 5.1;
- the decision-tree-based querying algorithm described in Section 5.2.2 in which all the candidate stem/paradigm pairs are assumed to have the same probability, that is, no feasibility score is used. The values initially defined by the ID3 algorithm for the probabilities $p(x)$ in Equation 2 and $p(t)$ in Equation 3 are used (that is, those based on the proportion of elements that belong to each class);
- the decision-tree-based querying algorithm in which each candidate stem/paradigm pair is scored with the feasibility score based on hidden Markov models as described in Section 5.2.1.

In addition to the average number of questions posed to the oracle, the HMM probabilities and heuristic scores were compared by evaluating their success in assigning the highest score or probability to the correct stem/paradigm candidate.

6.1.1 Data

In order to build test sets that would be as realistic as possible, the words to be inserted were chosen from different stages of the actual development of the different Apertium monolingual dictionaries. The revision history of the dictionaries in the Subversion repository of the Apertium project[18] has made this approach possible. Given a pair of dictionary revisions (R_1, R_2), where R_1 is an earlier revision than R_2, the evaluation task consisted of adding to R_1 the entries[19] in R_2 but not in R_1 (i.e., the relative complement

[13] `http://www.meta-net.eu/`

[14] `http://sourceforge.net/p/apertium/svn/HEAD/tree/trunk/apertium-es-ca/`

[15] `http://sourceforge.net/p/apertium/svn/HEAD/tree/trunk/apertium-eu-es/`

[16] `http://sourceforge.net/p/apertium/svn/HEAD/tree/trunk/apertium-mlt-ara/`

[17] `http://sourceforge.net/p/apertium/svn/HEAD/tree/trunk/apertium-en-es/`

[18] `https://svn.code.sf.net/p/apertium/svn/`

[19] Recall that an entry is made of a stem and a paradigm. An entry can generate multiple word forms when it is expanded. Each word form also has morphological information attached, although the morphological information is not included in most of the examples presented in this paper for the sake of simplicity.

of R_1 in R_2), which will be henceforth called target entries. In order to ensure that all the paradigms assigned to these entries were also available in R_1, all the revisions of the dictionary were sequentially checked and grouped according to their paradigm sets, thus obtaining ranges of *compatible revisions*. The number of new entries added between the oldest and newest revisions of each range were then computed, and a set of revision pairs obtained from among those with the greatest number of different entries were manually selected for the experiments.

Recall that a context sentence is needed for each word form to be inserted. These sentences were extracted from different parallel corpora:[20] News Commentary (Bojar et al, 2013) for Spanish, the Basque Public Administration Institute translation memory[21] for Basque, the corpus from the official journal of the Catalan Goverment (Tiedemann, 2012) for Catalan, and the DGT translation memory (Steinberger et al, 2014) for Maltese. A set of 174,441 segments were randomly extracted from each of these corpora.[22] Each corpus was then split into two parts: the first one, which contained 90% of the sentences, was used for training the HMM; the second one, which contained the remaining 10%, was used to extract the context sentences of the word forms of the test sets. For each revision pair, the new entries were expanded in order to obtain the corresponding word forms and, for each word form and sentence in which it could be found, the corresponding word form/context sentence pair was added to the test set associated with the revision pair and language. Table 3 shows the list of revision pairs, the total number of paradigms defined in both revisions (the paradigms are the same in both cases), the number of new target entries added to R_1 and the number of word forms included in the evaluation. With regard to the training of the HMM on 90% of the corpus, a different HMM was trained for each revision pair as described in Section 5.2.1. In all cases, the Baum–Welch algorithm was stopped after 9 iterations.[23]

The Wikipedia dumps for Spanish,[24] Catalan,[25] Basque[26] and Maltese[27] were used as the monolingual corpora to compute the feasibility scores in the heuristic-based approach in Section 5.1. The value of the threshold Θ used to compute the set $\text{Unusual}_C(c_n)$, also described in Section 5.1, was set to 0.1.[28]

6.2 Results

Table 4 shows, for each of the three systems and four languages evaluated, the average number of questions needed to determine the correct paradigm for the word forms

[20] Parallel corpora were chosen, instead of monolingual ones, simply because they are already segmented into sentences.

[21] `http://opendata.euskadi.eus/w79-contdata/es/contenidos/ds_recursos_linguisticos/memorias_traduccion/es_izo/index.shtml`

[22] This number was chosen since it corresponds to the size of the smallest corpus: that used for Spanish.

[23] This number of iterations was optimal in our preliminary experiments for Spanish; it was therefore used for the remaining languages in order to ensure the same experimental conditions.

[24] `http://dumps.wikimedia.org/eswiki/20110114/eswiki-20110114-pages-articles.xml.bz2`

[25] `https://dumps.wikimedia.org/cawiki/20150807/cawiki-20150807-pages-articles.xml.bz2`

[26] `https://dumps.wikimedia.org/euwiki/20150807/euwiki-20150807-pages-articles.xml.bz2`

[27] `https://dumps.wikimedia.org/mtwiki/20150806/mtwiki-20150806-pages-articles.xml.bz2`

[28] Preliminary experiments showed that this value of Θ caused the desirable effect of the most infrequent inflected forms not being taken into account (such as unusual combinations of enclitic pronouns in Spanish).

Table 3: Revision pairs of the Spanish, Catalan, Basque and Maltese monolingual dictionaries (belonging to the Apertium English–Spanish, Spanish–Catalan, Basque–Spanish, and Maltese–Arabic MT systems, respectively) used in the experiments, total number of paradigms declared, number of new entries (not in R_1 but included in R_2), and number of word forms found in the test section of the corpus.

Language	Revision pair		Number	Target	Word forms
	R_1	R_2	of paradigms	entries	in corpus
Spanish	7217	7287	456	109	485
	11762	12415	467	1802	550
	17582	20212	475	700	362
	27241	27627	492	1048	297
	34649	35985	494	1194	79
	36838	44118	494	1039	650
Catalan	6557	6917	522	1842	184
	7216	7269	522	580	55
	11653	12060	552	446	1414
Basque	1376	1410	292	106	4089
	5188	5290	257	7408	132
	5333	5596	271	115	280
Maltese	38228	38375	703	50170	6
	39468	39509	741	38970	84
	40109	40367	751	55679	153

in the test set (lower values represent better results). A cell in bold means that the corresponding method either outperforms or underperforms the other two methods by a statistically significant margin ($p \leq 0.05$).[29] If it outperforms them, the value in the cell is marked with the symbol $\uparrow$, whereas if it underperforms them, the value is marked with $\downarrow$.

It can be clearly observed that the system that uses the new probabilistic approaches for the feasibility score and querying algorithms needs fewer questions than the heuristic-based system for most of the test sets; only a few exceptions could be found, namely $(7217, 7287)$ for Spanish, $(7216, 7269)$ for Catalan, and $(1376, 1410)$ for Basque. In addition, the difference in the number of questions posed is statistically significant in all the test sets with the exception of the revision pair $(5333, 5596)$ for Basque. It is worth noting that in the exceptional cases in which the heuristic method proved to be better, the difference in the number of questions posed is relatively small.

It is also worth remarking how the decision tree without feasibility scores behaves: it is able to outperform the heuristic system in nine out of the fifteen test sets, even though it does not use any kind of feasibility score; this confirms its robustness and proves that a decision tree is more efficient (in terms of the number of questions posed)

[29] Statistical significance tests were performed with the randomisation version of the paired sample t test described by Yeh (2000) using the software available at `http://www.nlpado.de/~sebastian/software/sigf.shtml`.

Table 4: Average number of polar questions needed by the three approaches under evaluation (ID3-trained decision tree using HMM probabilities, ID3-trained decision tree using proportions, and heuristic-based approach) for each of the Spanish, Catalan, Basque, and Maltese test sets. A cell in bold means that the corresponding system either outperforms or underperforms the other two systems by a statistically significant margin ($p \leq 0.05$). If it outperforms them, the value in the cell is marked with the symbol $\uparrow$, whereas if it underperforms them, the value is marked with $\downarrow$.

Language	Revision pair		Average number of questions		
	R_1	R_2	ID3+HMM	ID3	Heuristic
Spanish	7217	7287	3.26	**5.50**$^{\downarrow}$	**3.08**$^{\uparrow}$
	11762	12415	5.22	5.26	**10.71**$^{\downarrow}$
	17582	20212	**4.74**$^{\uparrow}$	**5.65**$^{\downarrow}$	5.18
	27241	27627	**4.35**$^{\uparrow}$	5.72	5.85
	34649	35985	6.22	6.32	**8.67**$^{\downarrow}$
	36838	44118	**5.83**$^{\uparrow}$	6.11	**7.48**$^{\downarrow}$
Catalan	6557	6917	**9.41**$^{\uparrow}$	10.10	**11.97**$^{\downarrow}$
	7216	7269	3.85	**7.70**$^{\downarrow}$	**3.18**$^{\uparrow}$
	11653	12060	**8.11**$^{\uparrow}$	9.39	**27.73**$^{\downarrow}$
Basque	1376	1410	**8.25**$^{\downarrow}$	8.07	**5.87**$^{\uparrow}$
	5188	5290	**4.24**$^{\uparrow}$	7.43	**11.63**$^{\downarrow}$
	5333	5596	3.26	**7.13**$^{\downarrow}$	3.38
Maltese	38228	38375	14.50	14.50	**17.50**$^{\downarrow}$
	39468	39509	22.28	**22.00**$^{\uparrow}$	**39.00**$^{\downarrow}$
	40109	40367	15.41	**14.97**$^{\uparrow}$	**21.72**$^{\downarrow}$

than the previous heuristic querying algorithm. In addition, the results also confirm the successful integration of the feasibility scores computed with an HMM into the decision tree: when feasibility scores are used to compute the probabilities $p(x)$ and $p(t)$ defined in Section 5.2.2, the number of questions posed is reduced in all the test sets for Spanish and Catalan (the difference is statistically significant in four of the test sets for Spanish). For Basque, it only underperforms in one of the cases, while it is, in general, worse for Maltese. The most lileky explanation for this situation is that the more mature Spanish and Catalan dictionaries allowed us to train more reliable HMM models. On the contrary, the Maltese dictionary, which is the most undeveloped dictionary in our data set, could hardly take advantage of this source of information. The conclusion may therefore be that the strategy using an ID3 decision tree without feasibility scores is better for underdeveloped dictionaries.

However, the fact that adding an HMM-based feasibility score to the decision tree leads to a reduction in the number of questions for some language pairs does not necessarily mean that the HMM-based feasibility score is more accurate than the heuristic one. In order to clarify this issue, Table 5 shows, in the case of the Spanish dictionary,[30]

[30]In this analysis, we have chosen Spanish as a representative of the four languages evaluated and omitted

Table 5: Average position (starting from zero) of the correct paradigm in the list of candidate stem/paradigm pairs sorted by feasibility score and percentage of words in the Spanish test set for which the correct candidate is the first one for the HMM-based and heuristic feasibility scores. A cell in bold means that the corresponding system outperforms the other system by a statistically significant margin ($p \leq 0.05$). The last column represents the proportion of words in the test set for which none of the word forms generated by expanding the correct stem/paradigm combination can be found in the monolingual corpus used to compute the heuristic feasibility score.

Revision pair		Average position		% correct is first		% test words with
R_1	R_2	HMM	Heuristic	HMM	Heuristic	no evidence
7217	7287	1.47	**0.51**	70.31	**72.99**	0.20
11762	12415	**5.66**	10.45	**28.00**	8.36	54.36
17582	20212	1.87	1.72	**52.49**	40.88	0.00
27241	27627	7.11	**4.67**	39.73	**42.76**	9.66
34649	35985	6.66	**5.18**	45.57	45.57	37.79
36838	44118	**1.08**	3.51	**81.08**	70.52	2.10

the average position of the correct paradigm in the sorted candidate list, along with the percentage of words in the test set for which the correct paradigm was ranked first for both types of feasibility score. It also shows the fraction of words for which none of the word forms generated by expanding the correct stem/paradigm combination can be found in the monolingual corpus used to compute the heuristic feasibility score. The results vary across the different test sets: for some of them the HMM-based feasibility score is more accurate than the heuristic one, but for others it is the other way around.

The test set extracted from the revision pair $(11762, 12415)$ confirms that the HMM-based feasibility score is helpful when the corpus does not contain sufficient evidence. For more than a half of the word forms in the test set, none of the word forms resulting from the inflection of the correct paradigm and the corresponding stem can be found in the monolingual corpus. As a consequence, the average position of the correct stem/paradigm pair in the list L sorted by feasibility score is very high. The HMM-based feasibility score does not suffer as regards this issue, and the position of the correct stem/paradigm pair in the list sorted by HMM-based feasibility score is much closer to the first positions. In summary, the HMM-based feasibility score is able to find evidence in situations in which the heuristic one is not able to find it, but this does not mean that it is generally more reliable than the heuristic one. Since they extract a different type of information from the monolingual corpus, these results suggest that the two feasibility scores are complementary and could be combined in the future in order to make the most of the monolingual corpora available.

Another interesting conclusion that can be drawn from Table 4 and 5 is the confirmation of the robustness of the decision tree querying algorithm. In test sets $(27241, 27627)$ and $(34649, 35985)$, the fully probabilistic system is able to outperform the heuristic-based one despite the fact that the heuristic feasibility score is more accurate. When

the discussion for the remainder of them as the results are consistent across the different languages.

the heuristic feasibility score is less accurate, however, the number of questions posed by the heuristic system grows, as can be observed in test sets $(11762, 12415)$ and $(36838, 44118)$.

In conclusion, it has been proved that the probabilistic alternatives reduce the number of questions needed to be posed to users in order to insert new entries into a monolingual dictionary. Using a decision tree instead of the heuristic querying algorithm reduces the number of questions in almost all scenarios, but especially when the feasibility score is not accurate, while the HMM-based feasibility score seems to be complementary with the heuristic one.

7 Human evaluation

After the automatic evaluation presented in the previous section, the human evaluation discussed in this section confirms that, by using our method, new entries can be successfully inserted into monolingual dictionaries by non-expert users with high success rates. The experiment carried out consists of an evaluation involving real users in which the heuristic approach for the computation of the feasibility score and the querying algorithm have been followed (see Section 5.1).

7.1 Experimental set-up for the human evaluation

In order to confirm that average speakers of a language can successfully use the strategy described in Section 4 to insert new entries in a morphological dictionary, a group of 4 human evaluators (computer engineers without advanced linguistic knowledge) was chosen and asked to add a set of words to a monolingual dictionary of the Apertium rule-based MT platform (Forcada et al, 2011). The basic idea underlying our evaluation strategy is to choose a set of common words from a morphological dictionary, remove them from the dictionary, and ask the selected group of non-expert users to insert them using our method.

7.1.1 Data

The Apertium Spanish monolingual dictionary of the Spanish–Catalan[31] language pair was chosen as the dictionary in which the new entries needed to be inserted. Note that Spanish has been chosen here, instead of any of the under-resourced languages used in the experiments described in Section 6.1, since the authors had better access to regular speakers of this language and because the viability of the methods described for under-resourced languages has been already confirmed in previous experiments.

A Spanish Wikipedia dump[32] was chosen as the monolingual corpus used to compute the feasibility scores. The value of the threshold Θ used to compute the set $\text{Unusual}_C(c_n)$ was set at 0.1 as in the experiments in Section 6. In order to build the test set (i.e. the collection of word forms to be added by the users), the paradigms which meet the following restrictions were first selected:

[31] Revision 33900 in the repository `https://svn.code.sf.net/p/apertium/svn/trunk/apertium-es-ca`

[32] `http://dumps.wikimedia.org/eswiki/20110114/eswiki-20110114-pages-articles.xml.bz2`

- They belong to an open part-of-speech category. When creating the monolingual dictionary for a given language, words belonging to closed part-of-speech categories constitute a small set which should have already been completely inserted by expert users.

- After removing 5 words for their inclusion in the test set, at least one word form of one of the remaining words can be found in the monolingual corpus. This is needed to be able to properly compute $\text{Unusual}_C(c_n)$.

The 30 paradigms assigned to the 30 word entries whose inflected word forms have the highest aggregated frequency in the monolingual corpus were then chosen from among the paradigms fulfilling the previous conditions.[33] From each of these paradigms, the 5 most common words were extracted and a set with 150 words was built. For each of the words selected, the most common word form was chosen in order to ensure that users were familiar with the word forms to be added. These 150 word forms were used to build 4 subsets, one for each of the four non-expert human evaluators, introducing some redundancy in order to be able to compute inter-annotator and intra-annotator agreements. Each subset contained 50 word forms and was built as follows:

- 30 word forms were extracted from the initial set of word forms and they were not shared with any other evaluator. 120 word forms were consequently used in this way.

- 5 word forms randomly chosen from these 30 word forms were included twice in order to compute intra-annotator agreement.

- Of the remaining 30 word forms in the initial set of 150 word forms, 5 word forms were assigned to each pair of evaluators in order to compute inter-annotator agreement (there were 6 evaluator pairs, thus the 30 word forms were used). From the point of view of the test set assigned to each evaluator, 15 word forms were obtained in this way (5 words multiplied by 3 remaining evaluators).

A sentence randomly chosen from the monolingual corpus containing the word form to be classified was also shown to the user. As previously stated, the strategy presented in this paper is meant to be applied when a user of an MT system translates a sentence that contains a word that is not present in its dictionaries. In the proposed evaluation scenario, the sentence additionally helps to ease the classification of homographs.

7.1.2 Evaluation metrics

In order to estimate the reliability of the results, pair-wise inter-annotator agreement for each pair of annotators and intra-annotator agreement for each annotator were computed using Cohen's κ (Cohen, 1960). In addition, the following evaluation scores were calculated:

[33] Repeated paradigms are excluded; for instance, if the first two entries with the highest frequency belong to the same paradigm, the second one is replaced with another paradigm.

- Success rate: percentage of word forms in the test set that were tagged with the paradigm originally assigned to them in the monolingual dictionary.
- Average precision and recall: precision (P) and recall (R) were computed as

$$P(c,c') = \frac{|I(c) \cap I(c')|}{|I(c)|} \qquad R(c,c') = \frac{|I(c) \cap I(c')|}{|I(c')|},$$

  where c is the stem/paradigm pair obtained and c' is the stem/paradigm pair originally found in the dictionary. This metric is intended to assess the similarity between the chosen paradigm and that originally present in the dictionary.
- Average position of the correct candidate in the initial sorted list L of stem/paradigm pairs, which provides an estimation of the accuracy of the feasibility score.
- Average number of questions posed by the system to the user for each word.

These metrics (except for the last one) were also computed for a non-interactive baseline in which the chosen paradigm was that with the highest feasibility score, so as to assess whether the feasibility score computed from a monolingual corpus is sufficient to correctly choose the correct paradigm.

7.2 Results of the human evaluation

Before describing and discussing the results obtained in the human evaluation, it is worth assessing their reliability by analysing the values observed for the pair-wise inter-annotator agreement of each pair of human evaluators (shown in Table 6a) and the intra-annotator agreement (Table 6b). According to Cohen (1960), a value of κ between 0.6 and 0.8 is usually interpreted as a *good agreement*, and when it ranges between 0.8 and 1.0 it is usually stated that there is a *very good agreement* between annotators. Since all the values obtained fall in one of these ranges, it can be concluded that each of the 4 annotators was quite consistent (high intra-annotator agreement) and that they agreed on their answers (high inter-annotator agreement), which ensures confidence on the remaining results.

Table 7 shows the value of the five evaluation metrics for our interactive framework using the heuristics defined in Sections 5.1.1 and 5.1.2, and for a non-interactive baseline method that consists of simply choosing the candidate paradigm/stem with the highest feasibility score. Confidence intervals were estimated with 95% statistical significance with a *t-test*. Results show high success rates, and this confirms that users are able to correctly answer the polar questions posed by the system and, consequently, insert the corresponding entry into the morphological dictionary. The difference between this and the non-interactive baseline is remarkable: the feasibility score itself is not as accurate as the users' answers as regards correctly assigning paradigms to new words. The recall of the baseline, however, is relatively high, which suggests that the paradigms chosen by the non-interactive approach generate many of the correct word forms, but also many incorrect ones. In addition, the values for precision and recall (around 95%, higher than the success rate) for the interactive approach suggest that those words which were assigned to incorrect paradigms, were assigned to paradigms

Table 6: Inter-annotator agreement (a) and intra-annotator agreement (b) computed using Cohen's κ in the experiments with a Spanish dictionary involving the heuristic feasibility score and querying algorithm described in Sections 5.1.1 and 5.1.2.

Annotator pair	κ
A–B	0.76
A–C	0.74
A–D	0.71
B–C	0.76
B–D	1.00
C–D	1.00
average	0.83

(a)

Annotator	κ
A	1.00
B	0.73
C	1.00
D	1.00
average	0.93

(b)

Table 7: Success rate, precision, recall, initial position of the correct paradigm/stem in L and average number of questions posed to the evaluators (95% statistical significance) when inserting entries into the Apertium Spanish monolingual dictionary using our interactive method with the heuristic feasibility score and querying algorithm described in Sections 5.1.1 and 5.1.2, and using a non-interactive baseline in which the c_n pair with highest feasibility score is automatically chosen.

System	success rate	P	R	initial position in L	# questions
non-interactive baseline	$16\% \pm 5$	$49\% \pm 5$	$88\% \pm 3$	-	-
interactive heuristic method	$88\% \pm 5$	$94\% \pm 3$	$95\% \pm 3$	12 ± 2	6.1 ± 0.7

that share many word forms with the correct one. This constitutes a clear advantage when the dictionary that contains the entries inserted by the non-expert users are used in an MT system: even words for which the most appropriate paradigm has not been chosen can make the system analyse and translate word forms that it was not possible to analyse before.[34]

Some of the most common mistakes made by the users were related to verbs and superlative adjectives. Spanish morphological rules allow multiple concatenations of enclitic pronouns at the end of verbs. On many occasions, users rejected forms of verbs with too many enclitic pronouns or for which some concrete enclitics had no semantic meaning (for instance, *viájasela*). This occurs because, in order to reduce the number of possible paradigms, Apertium's dictionaries can assign some words to existing paradigms which are a superset of the correct one; since the semantically incorrect word forms included will never occur in a text to be translated, this may, in princi-

[34]These results are compatible with those obtained in a previous evaluation (Esplà-Gomis et al, 2011), in which the test set was obtained by randomly picking a pair of words from 166 different paradigms; in that case, 10 non-expert humans evaluators took part, and annotator agreement metrics were not computed. In those experiments, the value of average precision and recall was slightly below 90%, although the recall of the non-interactive baseline was much lower.

ple, be done safely. Regarding superlative adjectives, Apertium contains paradigms for adjectives which have superlative form and for those which do not. The users often accepted the superlative form of adjectives which do not have one.

8 Conclusions and future work

In this paper, a new method with which to allow non-expert users insert new entries into the morphological dictionaries used in rule-based MT has been presented, although the approach can easily be extended to the morphological dictionaries used in many other natural language processing applications. The involvement of non-expert users is particularly critical for under-resourced languages which either lack or cannot afford the recruitment of skilled workforce. It has been proved that non-expert users are able to successfully validate whether certain word forms are valid forms of the word to be inserted. Our method creates the corresponding entry in the morphological dictionary from the answers provided by the users with the help of existing inflection paradigms. The correct entry (stem and inflection paradigm) was inserted in the case of most of the words that users were asked to add to the dictionary during the evaluation process. Moreover, when the inserted entry was not the correct one, it often shared most of its inflected word forms with the correct one, thus still increasing the coverage of the system. The use of a binary decision tree to decide which word forms need to be validated by the users ensures that the task is performed efficiently: with the exception of the case of Maltese, only 4–9 questions were usually needed in order to insert a set of words selected from the revision histories of real Apertium monolingual dictionaries. The Java code for the resulting system is available[35] under the free/open-source GNU General Public License.[36]

Given that it has been proved that the strategy achieves good results with real users and in the automatic evaluation, this work opens up new opportunities for the cheap enlargement of morphological dictionaries when collaborators who have mastered the particular encoding of the dictionary are not available or, even if they are, in situations in which this approach allows them to focus on the development of more complex parts of the system while users with less experience carry out the task. As already pointed out, this approach could be integrated into a rule-based MT system and users could be asked to help insert the words into the sentences to be translated that are not found in the system's dictionaries. Moreover, users could also be contacted by means of a crowdsourcing platform (Wang et al, 2013).

Finally, it is worth noting that, as stated in Section 4.2, the approach presented in this paper is not able to choose from among paradigms that generate the same set of word forms but with different associated morphological information. For example, in the experiments described in Section 6 for Spanish, the final solution contained more than one paradigm with the same word forms for around 87% of word forms in the test sets. On average, the final solution contained 8.35 paradigms, while the average number of candidate stem/paradigm pairs was 29.77. This signifies that in order to

[35] `https://svn.code.sf.net/p/apertium/svn/branches/dictionary-enlargement`

[36] `http://www.gnu.org/licenses/gpl.html`

use this approach without the intervention of expert users, either the missing morphological information needs to be elicited by asking the user different questions, or the paradigm with the most appropriate morphological information needs to be selected in a fully automatic manner. Nevertheless, our system can be used out-of-the-box if the experts choose the most appropriate morphological information at the end of the process: choosing from among 8.35 candidate stem/paradigm pairs is definitely easier and faster than choosing from among 29.77. Other future research directions include the use of the feasibility score computed by employing an HMM to select the most appropriate paradigm from those that generate exactly the same word forms, and combining the heuristic and HMM-based feasibility scores, which seem complementary according to the experimental results described in Section 6.2. Another interesting way in which to improve our approach may be the adoption of methods that can be used to reduce the amount of word forms posed to the user by discarding candidates that generate word forms that are unlikely, or even impossible in a given language. This could be done by, for example, using phonotactic models such as those proposed by Belz (2000). These models learn the classes of syllabes of a given language and how they can be combined using finite-state automata. Such models would allow us to detect those combinations of stem and suffix that make no sense in a given language, thus removing some candidates before starting the interaction with the user.

Acknowledgements

This work has been partially funded by the Spanish Ministry of Science & Innovation through project TIN2009-14009-C02-01, by the Spanish Ministry of Economy & Competitiveness through project TIN2012-32615, by the Generalitat Valenciana through grant ACIF/2010/174 from VALi+d programme, and by the European Commission through project PIAP-GA-2012-324414 (Abu-MaTran).

References

Ahlberg M, Forsberg M, Hulden M (2014) Semi-supervised learning of morphological paradigms and lexicons. In: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, Gothenburg, Sweden, pp 569–578

Ambati V, Vogel S, Carbonell J (2010) Active learning and crowd-sourcing for machine translation. In: Proceedings of the 7th International Conference on Language Resources and Evaluation, Valletta, Malta, LREC'10, pp 2169–2174

Bartusková D, Sedlácek R (2002) Tools for semi-automatic assignment of Czech nouns to declination patterns. In: Proceedings of the 5th International Conference on Text, Speech and Dialogue, Brno, Czech Republic, pp 159–164

Baum LE (1972) An inequality and associated maximization technique in statistical estimation for probabilistic functions of a Markov process. Inequalities 3:1–8

Belz A (2000) Multi-syllable phonotactic modelling. In: Finite-State Phonology: Proceedings of the 5th Workshop of the ACL Special Interest Group in Computational Phonology, Luxembourg, pp 569–578

Bojar O, Buck C, Callison-Burch C, Federmann C, Haddow B, Koehn P, Monz C, Post M, Soricut R, Specia L (2013) Findings of the 2013 Workshop on Statistical Machine Translation. In: Proceedings of the 8th Workshop on Statistical Machine Translation, pp 1–44

Cohen J (1960) A coefficient of agreement for nominal scales. Educational and Psychological Measurement 20(1):37–46

Cristianini N, Shawe-Taylor J (2000) An introduction to support vector machines and other kernel-based learning methods. Cambridge University Press

Cutting D, Kupiec J, Pedersen J, Sibun P (1992) A practical part-of-speech tagger. In: Proceedings of the 3rd Conference on Applied Natural Language Processing, pp 133–140

Desai S, Pawar J, Bhattacharyya P (2012) Automated paradigm selection for FSA based Konkani verb morphological analyzer. In: 24th International Conference on Computational Linguistics: Demonstration Papers, pp 103–110

Détrez G, Ranta A (2012) Smart paradigms and the predictability and complexity of inflectional morphology. In: Proceedings of EACL, pp 645–653

Esplà-Gomis M, Sánchez-Cartagena VM, Pérez-Ortiz JA (2011) Enlarging monolingual dictionaries for machine translation with active learning and non-expert users. In: Proceedings of the International Conference Recent Advances in Natural Language Processing 2011, Hissar, Bulgaria, pp 339–346

Esplà-Gomis M, Sánchez-Cartagena VM, Sánchez-Martínez F, Carrasco RC, Forcada ML, Pérez-Ortiz JA (2014) An efficient method to assist non-expert users in extending dictionaries by assigning stems and inflectional paradigms to unknown words. In: Proceedings of the 17th Annual Conference of the European Association for Machine Translation, pp 19–26

Font-Llitjós A (2007) Automatic Improvement of Machine Translation Systems. PhD thesis, Carnegie Mellon University

Forcada ML, Ginestí-Rosell M, Nordfalk J, O'Regan J, Ortiz-Rojas S, Pérez-Ortiz JA, Sánchez-Martínez F, Ramírez-Sánchez G, Tyers FM (2011) Apertium: a free/open-source platform for rule-based machine translation. Machine Translation 25(2):127–144, special Issue: Free/Open-Source Machine Translation

Haffari G, Roy M, Sarkar A (2009) Active learning for statistical phrase-based machine translation. In: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Boulder, Colorado, NAACL '09, pp 415–423

Hutchins WJ, Somers HL (1992) An introduction to machine translation, vol 362. Academic Press New York

Kilbury J, Naerger P, Renz I (1992) Lexical entries for unknown words. Theorie des Lexikons:Arbeiten des Sonderforschungsbereichs 282(29)

Koehn P (2010) Statistical Machine Translation. Cambridge University Press

McCreight EM (1976) A space-economical suffix tree construction algorithm. Journal of the Association for Computing Machinery 23(2):262–272

McShane M, Nirenburg S, Cowie J, Zacharski R (2002) Embedding knowledge elicitation and MT systems within a single architecture. Machine Translation 17:271–305

Monson C (2009) ParaMor: From Paradigm Structure to Natural Language Morphology Induction. PhD thesis, Carnegie Mellon University

Quinlan JR (1986) Induction of decision trees. Machine Learning 1(1):81–106

Rabiner L (1989) A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the Institute of Electrical and Electronics Engineers (IEEE) 77(2):257–286

Rehm G, Uszkoreit H (2013) META-NET Strategic Research Agenda for Multilingual Europe 2020. Springer Berlin Heidelberg

Sánchez-Cartagena VM, Esplà-Gomis M, Sánchez-Martínez F, Pérez-Ortiz JA (2012a) Choosing the correct paradigm for unknown words in rule-based machine translation systems. In: Proceedings of the 3rd International Workshop on Free/Open-Source Rule-Based Machine Translation, Gothenburg, Sweden, pp 27–39

Sánchez-Cartagena VM, Esplà-Gomis M, Pérez-Ortiz JA (2012b) Source-language dictionaries help non-expert users to enlarge target-language dictionaries for machine translation. In: Proceedings of the 8th International Conference on Language Resources and Evaluation, Istanbul, Turkey, LREC'12, pp 3422–3429

Šnajder J (2013) Models for Predicting the Inflectional Paradigm of Croatian Words. Slovenščina 20: empirical, applied and interdisciplinary research 1(2):1–34

Steinberger R, Ebrahim M, Poulis A, Carrasco-Benitez M, Schlüter P, Przybyszewski M, Gilbro S (2014) An overview of the European Union's highly multilingual parallel corpora. Language Resources and Evaluation 48(4):679–707

Tiedemann J (2012) Parallel data, tools and interfaces in opus. In: Proceedings of the 8th International Conference on Language Resources and Evaluation, Istanbul, Turkey, LREC'12, pp 2214–2218

Wang A, Hoang C, Kan M (2013) Perspectives on crowdsourcing annotations for natural language processing. Language Resources and Evaluation 47(1):9–31

Yeh A (2000) More accurate tests for the statistical significance of result differences. In: Proceedings of the 18th Conference on Computational Linguistics - Volume 2, Stroudsburg, USA, COLING '00, pp 947–953