An efficient method to assist non-expert users in extending dictionaries by assigning stems and inflectional paradigms to unknown words

Miquel Esplà-Gomis mespla@dlsi.ua.es
Víctor M. Sánchez-Cartagena vmsanchez@dlsi.ua.es
Felipe Sánchez-Martínez fsanchez@dlsi.ua.es
Rafael C. Carrasco carrasco@dlsi.ua.es
Mikel L. Forcada mlf@dlsi.ua.es
Juan Antonio Pérez-Ortiz japerez@dlsi.ua.es

Dept. de Llenguatges i Sistemes Informàtics, Universitat d'Alacant, 03071 Alacant, Spain

Abstract

A method is presented to assist users with no background in linguistics in adding the unknown words in a text to monolingual dictionaries such as those used in rule-based machine translation systems. Adding a word to these dictionaries requires identifying its stem and the inflection paradigm to be used in order to generate all its word forms. Our method is based on a previous interactive approach in which non-expert users were asked to validate whether some tentative word forms were correct forms of the new word; these validations were then used to determine the most appropriate stem and paradigm. The previous approach was based on a set of intuitive heuristics designed both to obtain an estimate of the eligibility of each candidate stem/paradigm combination and to determine the word form to be validated at each step. Our new approach however uses formal models for both tasks (a hidden Markov model to estimate eligibility and a decision tree to select the word form) and achieves significantly better results.

1 Introduction

Creation of the linguistic data (such as monolingual dictionaries, bilingual dictionaries, transfer rules, etc.) required by rule-based machine translation (RBMT) systems has usually involved teams of trained linguists. However, development costs could be significantly reduced by involving a broader group of non-expert users in the extension of these resources. This may include, for instance, the very same users of the machine translation (MT) system or accidental collaborators recruited through crowdsourcing platforms (Wang et al., 2013). The scenario considered in this paper is that of non-expert *users* (in a general sense) who have to introduce into the two monolingual dictionaries[1] of a RBMT system the unknown words found in an input text so that the system is subsequently able to correctly translate them.[2] Note, however, that our method could be applied to the addition of entries into the morphological dictionaries used in many other natural language processing applications. 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 inserted as well.

Inflection paradigms are commonly introduced in RBMT systems in order to group regularities in the inflection of a set of words;[3] 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,[4]

[1] One source-language dictionary used for morphological analysis and one target-language dictionary used for morphological generation.

[2] It could also happen that the word form is not completely unknown, but it is assigned to a different paradigm; for example, the word *fly* could already be included in a dictionary as a verb, but a user may need to insert it as a noun.

[3] Paradigms ease the management of dictionaries in two ways: 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.

[4] The stem is the part of a word that is common to all its

the gerund is obtained; by adding the suffix *-ed*, the past is obtained; etc. Adding a new entry to a monolingual dictionary therefore implies determining the stem of the new word and a suitable inflection paradigm among those defined by the MT system for the corresponding language. In this work we assume that the paradigms for all possible words in the language are already included in the dictionary.[5] We will focus on monolingual dictionaries because insertion of information in the bilingual dictionaries of RBMT systems is usually straightforward (Sánchez-Cartagena et al., 2012a).

Our approach improves a previous interactive method (Esplà-Gomis et al., 2011) that was based on a number of intuitive heuristics; the improvement presented here is twofold: on the one hand, more coherent and principled models are introduced; on the other hand, the results are significantly better.

The rest of the paper is organised as follows. Section 2 discusses other works related to our proposal. Section 3 introduces the concepts on monolingual dictionaries that will be used in the remainder of the paper. An overview of the previous method (Esplà-Gomis et al., 2011) for dictionary extension is presented in Section 4, followed by the description of our new approach in Section 5. Section 6 discusses our experimental setting in which a Spanish monolingual dictionary is used, while the results obtained are presented and discussed in Section 7. Finally, some concluding remarks are presented in Section 8.

2 Related work

In this section, related works in literature are commented and compared with the common features in our new approach and in the work by Esplà-Gomis et al. (2011).

Two of the most prominent works in literature in relation to the elicitation of knowledge to build or improve RBMT systems are those by Font-Llitjós (2007) and McShane et al. (2002). The former proposes a strategy for improving both transfer rules and dictionaries by analysing the postediting process performed by a non-expert user through a dedicated interface. McShane et al. (2002) design a framework 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 system provides users with a lot of information about different linguistic phenomena to ease the elicitation task. Unlike these two approaches, our method is aimed at transfer-based MT systems in which a single translation is generated and no language model is used in order to rank a number of translation hypothesis; this kind of systems are notably more sensitive to erroneous linguistic information. We also want to relieve users from acquiring 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 right inflection paradigm for a new word to be inserted in an MT system dictionary. 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, in some cases, some morphological inflection. There are two important differences with our approach: firstly, smart paradigms are created exclusively by human experts; and secondly, users of smart paradigms need to have some linguistic background. For instance, an expert could decide that in order to correctly choose the inflection paradigm of most verbs in French the infinitive and the first person plural present indicative forms are needed; dictionary developers must then provide these two forms when inserting a new verb. Bartusková and Sedlácek (2002) also present a tool for 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 verbs and present a system for paradigm assignment based on the 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 turns the choice of the most appropriate paradigm for a given word into 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* classifier. The main difference with our approach

inflected forms.

[5]This can be easily expected as most unknown words belong to regular paradigms.

lies in the fact that their method is designed to be used in a fully-automatic pipeline, while we use the inferred models in order to minimise the number of queries posed to non-expert users. Finally, the automatic identification of morphological rules to segment a word into morphemes (a problem for which paradigm identification is a potential resolution strategy) has also been recently addressed (Monson, 2009; Walther and Nicolas, 2011).

3 Preliminaries

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[6] which is appended to stems to build new *word forms*, and m_{ij} is the corresponding morphological information. Given a *stem/paradigm* pair $c = t/p_i$ 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 in p_i to t. For instance, an English dictionary may contain the stem *want-* assigned to a paradigm with suffixes[7] p_i = {-, *-s*, *-ed*, *-ing*}; 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, our objective is to find both the stem $t \in \Pr(w)$[8] and the paradigm p_i such that $I(\text{want}/p_i)$ is the set of word forms which are all the correct forms of the unknown word. To that end, a set L containing all the stem/paradigm pairs compatible with w is determined by using a *generalised suffix tree* (McCreight, 1976) containing all the possible suffixes included in the paradigms in P.

The following example illustrates the previous definitions. Consider a simple dictionary with only four paradigms: p_1 = {-, *-s*}; p_2 = {*-y*, *-ies*}; p_3 = {*-y*, *-ies*, *-ied*, *-ying*}; and p_4 = {*-a*, *-um*}. Let's assume that the new word form is w=*policies* (actually, the noun *policy*); the compatible stem/paradigm pairs which will be obtained after this stage are: c_1=*policies*/p_1; c_2=*policie*/p_1; c_3=*polic*/p_2; and c_4=*polic*/p_3.

[6] Although our approach focuses on languages generating word forms by adding suffixes to stems (for example, Romance languages), it could be easily adapted to inflectional languages based on different ways of adding morphemes.

[7] We hereinafter omit the morphological information contained in p_i and show only the suffixes.

[8] $\Pr(w)$ is the set of all possible prefixes of w.

4 Previous approach

Esplà-Gomis et al. (2011) have already proposed an interactive method for extending the dictionaries of RBMT systems with the collaboration of non-expert users. In their work, the most appropriate stem/paradigm pair is chosen by means of a sequence of simple *yes/no questions* whose answer only requires *speaker-level* understanding of the language. Basically, users are asked to validate whether some word forms resulting from tentatively assigning different compatible stem/paradigm pairs in L (see Section 3) to the new word are correct word forms of it. The specific forms that are presented to the users for validation are automatically obtained by estimating the most informative ones which allow the system to discard the greatest number of wrong candidate paradigms at each step. The results showed (Esplà-Gomis et al., 2011) that the average number of queries posed to the users for a Spanish monolingual dictionary was around 5, which is reasonably small considering that the average number of initially compatible paradigms was around 56. Furthermore, Sánchez-Cartagena et al. (2012a) have shown that when the source-language word has already been inserted, the system is able to more accurately predict the right target-language paradigm by exploiting the correlations between paradigms in both languages from the corresponding bilingual dictionary, thus reducing significantly the number of questions.

After obtaining the list of compatible stem/paradigm pairs L, the original method performs three tasks: stem/paradigm pair scoring, selection of word forms to be offered to the user for validation an discrimination between equivalent paradigms.

Paradigm scoring. A *feasibility score* is computed for each compatible stem/paradigm pair $c_n \in L$ using a large monolingual corpus C. Candidates producing a set of word forms which occur more frequently in the corpus get higher scores. Following our 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*}. Using a large English corpus, word forms *policies* and *policy* will be easily found, and the rest of them (*policie*, *policiess*, *policied* and *policying*) probably will not. Therefore, c_3 would probably obtain the highest feasibility score.

Selection of word forms. The best candidate is chosen from L by querying the user about a reduced

set of the word forms for some of the compatible stem/paradigm pairs $c_n \in L$. To do so, the system first sorts L in descending order using the feasibility score. Then, users are asked (following the order in L) to confirm whether some of the word forms in each compatible stem/paradigm pair are correct forms of w. In this way, when a word form w' is accepted by the user, all $c_n \in L$ for which $w' \notin I(c_n)$ are removed from L; otherwise, all $c_n \in L$ for which $w' \in I(c_n)$ are removed from L. In order to attempt to maximise the number of word forms discarded and consequently minimise the amount of yes/no questions, users are iteratively asked to validate the word form from the first compatible stem/paradigm pair in L which exists in the minimum number of other compatible stem/paradigm pairs. This process is repeated until only one candidate (or a set of equivalent candidates; see next) remains in L.

Equivalent paradigms. When more than one paradigm provides exactly the same set of suffixes but with different morphological information, no additional question can be asked in order to discriminate between them.[9] For example, in the case of Spanish, many adjectives such as *alto* ('high') and nouns such as *gato* ('cat') are inflected identically. Therefore, two paradigms producing the same collection of suffixes {*-o* (masculine, singular), *-a* (feminine, singular), *-os* (masculine, plural), *-as* (feminine, plural)} but with different morphological information are defined in the monolingual dictionary, the stems *alt-* and *gat-* assigned to one of them each. This issue also affects paradigms with the same lexical category: *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 yes/no question can consequently be asked in order to discriminate between both paradigms. Sánchez-Cartagena et al. (2012b) proposed a solution to this issue that consisted of introducing an n-gram-based model of lexical categories and inflection information which was used as a final step[10] to automatically choose the right stem/paradigm pair with success rates between 56% and 96%.

[9] Around 81% of the word forms in a Spanish dictionary have been reported (Sánchez-Cartagena et al., 2012b) to be assignable to more than one equivalent paradigm.

[10] Note that this model is disconnected from the models used for scoring the compatible paradigms and deciding the word forms to be shown to the user.

5 Method

The approach discussed in the preceding section provides a complete framework for dictionary extension, but this framework can still be improved if more rigorous and principled models rather than intuitive heuristics are used. We propose consequently to replace those heuristics with hidden Markov models (HMMs) (Rabiner, 1989) and binary decision trees as follows. For a given unknown word form, first the set L of compatible stem/paradigm pairs is determined (see Section 3). The probability of each of them is then estimated by means of a first-order HMM. After that, these probabilities are used in order to build a decision tree which is used to guide the selection of words to be offered to the non-expert user for validation. Note that, unlike in the original method in which isolated unknown words were inserted into the dictionary, the HMM in our new method explicitly considers the sentence in which the new word appears and uses this contextual information in order to better estimate the likelihood of each compatible stem/paradigm pair. The objective here is to minimise the interaction with the user so that the addition of new words is made as fast as possible.

Hidden Markov models. 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. The system produces an output each time a state is reached after a transition. In our method, Γ is made up of all the paradigms in the dictionary and Σ corresponds to the set of suffixes produced by all these paradigms.

Our HMMs are trained in a way very similar to HMMs used in 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 text corpus as follows: (i) the monolingual dictionary is used in order to obtain the set F of all possible word forms; (ii) the word forms in the text corpus that belong to F are assigned all their corresponding suffix and paradigm pairs; (iii) the word forms not in F are assigned the set of suffix and paradigm pairs obtained from the set L of their compatible candidates, as described in Section 4. Once the HMM is trained, the probability $q_t(c_n)$

of assigning the word form located at position t in the sentence to the compatible candidate $c_n \in L$ can be computed by applying the following equation, which corresponds to Eq. (27) in the tutorial by Rabiner (1989):

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

This equation computes the probability that the model is in state c_n when at position t. In the equation, $\alpha_t(c_n)$ accounts for the (forward) probability of the sub-sentence from the begining of the sentence to position t given state c_n at 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, given state c_n at position t (Rabiner, 1989).

Decision trees. Decision trees are commonly used to learn classifiers: the internal nodes (decision nodes) of a decision tree are labelled with an input feature, an arc coming from an internal node exists for each possible feature value, and leaves are labelled with classes. The ID3 algorithm (Quinlan, 1986) has been proposed in order to build these trees. This algorithm follows a greedy approach (the resulting trees are therefore sub-optimal) by selecting the most appropriate attribute to split the data set on each iteration. The algorithm starts from the root of the tree with the whole data set S. At each iteration, an attribute A is picked for splitting S, being A the attribute providing the highest information gain. A child node is then created for each possible value of A, with a new test set containing only the elements matching this attribute value. The information gain measures the difference in entropy before and after S is split; for computing this entropy, the probability of each class is approximated by using the *proportion* of elements belonging to each of them.

Our method uses ID3 in order to build a binary (each node corresponds to a yes/no question) decision tree for each new word. Each class corresponds to a compatible stem/paradigm and the attribute set is made up of the set of different word forms, i.e. $\cup_{c_i \in L} I(c_i)$. The entropy in the ID3 algorithm could in principle be computed as stated before, i.e. by using the proportion of word forms derived from every stem/paradigm combination. In our approach, however, a more accurate computation of the entropy is proposed by using the class probability provided by the hidden Markov model.

A weakness that this method shares with the one described in Section 4 is that candidate paradigms producing the same collection of suffixes cannot be differentiated with yes/no questions. Therefore, at the end of the querying process, it is possible for more than one candidate to remain. In order to deal with this, the already computed HMM contextual probabilities could be used rather than the additional n-gram model of morphological information proposed by Sánchez-Cartagena et al. (2012b).[11] For this work, as in the one by (Sánchez-Cartagena et al., 2012a), we considered these paradigms producing the same word forms as equivalent and, therefore, they count as a single paradigm.

6 Experimental Setting

In order to ensure an accurate comparison between the methods described in Sections 4 and 5, our experimental framework replaces non-expert users, to which this method is eventually addressed, with an oracle so that interferences caused by human errors are avoided. The evaluation consisted of simulating the addition of a set of words to the Spanish monolingual dictionary of the Spanish–Catalan Apertium MT system (Forcada et al., 2011).

Six test sets were built consisting of sentences in Spanish containing at least an unknown word. Using an oracle, the average number of questions needed in order to obtain the correct paradigm was computed for the following three methods: the original approach by Esplà-Gomis et al. (2011) described in Section 4, a decision tree using proportions rather than probabilities,[12] and a decision tree assigning the probabilities estimated by an HMM. It is worth noting that this metric ignores the fact that, depending on the word form posed, a user could need more time to decide whether to accept or reject it. This will be evaluated in a future work. In addition to the average number of questions, the HMM probabilities and the feasibility scores of the original approach were compared by evaluating the success in detecting the correct paradigm, that is, in assigning the highest score or probability to the correct paradigm. This second metric is aimed at measuring the relation between the relative correctness in the probability/score assignment and the number of queries posed to the user.

Each of the six data sets consists of (i) a mono-

[11] Although out of the scope of this work, it could be interesting to compare both approaches to the task of choosing (or supporting a user to choose) the best correct compatible stem/paradigm combination.

[12] As in this approach there is only one element per class, this is equivalent to consider all classes as equiprobable.

lingual dictionary D; (ii) a collection of text sentences S containing each at least one word form of a word not in D; and (iii) the list of the correct stem/paradigm combination for the target word forms to be added to the dictionary, which is used as the oracle for our evaluation.

In order to measure the feasibility of these methods at different times in the development of a dictionary, the revision history of the dictionary in the Apertium project Subversion repository was used.[13] This strategy also allowed us to use the different revisions in order to build the oracle for the experiments: given a pair of dictionary revisions (R_1, R_2) with R_1 being an earlier revision than R_2, the evaluation task consisted of adding to R_1 the words in R_2 but not in R_1 (i.e., the relative complement of R_1 in R_2), which will be called, henceforth, target words. In order to ensure that all the paradigms assigned to these words were also available in R_1, we sequentially checked all the revisions of the dictionary and grouped them according to their paradigm definitions, thus obtaining ranges of *compatible revisions*. We then computed the number of words differing between the oldest and newest revisions of each range, and manually picked for the experiments six revision pairs among those with the greatest number of different words.

In order to obtain sentences containing the target words, the Spanish side of the parallel corpus News Commentary (Bojar et al., 2013) was used.[14] The corpus was split into two parts, one containing 90% of the sentences, which were used for training the HMM, and another one including the remaining 10%, which were used for testing. Sentences not containing at least one word form of one of the target words were removed from each test set. Table 1 shows the list of revision pairs, the number of words differing between them and the number of word forms included in the evaluation text. For both the training and testing corpora, the text was processed by following the strategy described in Section 5 using the revision R_1 of each test set. A different HMM was therefore trained for each test set; in all cases, the Baum–Welch algorithm was stopped after 9 iterations.

Finally, following the experimental setting proposed by Sánchez-Cartagena et al. (2012a), a word list obtained from the Spanish Wikipedia dump[15] was used as the monolingual corpus to compute the the feasibility scores in the heuristic-based approach in Section 4.

Revision pair R_1	Revision pair R_2	Target words	Target word forms in corpus
7217	7287	109	485
11762	12415	1802	550
17582	20212	700	362
27241	27627	1048	297
34649	35985	1194	79
36838	44118	1039	650

Table 1: Revision pairs of the Spanish monolingual dictionary in the Apertium Spanish–Catalan MT system used in the experiments, number of target words (added from R_1 to R_2), and number of target word forms appearing in the corpus.

7 Results and Discussion

Table 2 shows the average number of questions needed to determine the correct paradigm for the target words evaluated. Since the objective of our method is to reduce the interaction with the user as much as possible, lower values represent better results. Cells in bold correspond to statistically significant differences between the corresponding method and the two other approaches with $p \leq 0.05$.[16] Those values which are significantly better are marked with the symbol $\uparrow$, whereas values significantly worse are marked with $\downarrow$. As can be seen, the two decision-tree-based approaches are, in general, better than the heuristic-based approach. Contrary evidence however is seen for the sole case of the test set corresponding to revision pair $(7217, 7287)$. Furthermore, using the HMM probabilities for computing information gain in the ID3 algorithm results in a statistically significant improvement to the original ID3 method in four out of the six test sets evaluated. In order to shed some light on these results, additional experiments were performed in order to check how well the feasibility scores and the HMM-based probabilistic model ranked the candidate paradigms. Table 3 shows the average position of the correct paradigm in the sorted candidate list, as well as the percentage of times that the correct paradigm was ranked as the first one. Overall, the results in this table suggest that the quality of the ranking has a higher impact

[13] `https://svn.code.sf.net/p/apertium/svn/trunk/apertium-en-es/apertium-en-es.es.dix`

[14] This corpus was chosen because it belongs to an heterogeneous domain and it is already segmented into sentences.

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

[16] Statistical significance tests were performed with *sigf*, available at `http://www.nlpado.de/~sebastian/software/sigf.shtml`

Revision pairs		mean number of queries		
R_1	R_2	**ID3+HMM**	**ID3**	**Original**
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**$^{\downarrow}$

Table 2: Mean number of yes/no questions needed by the tree approaches under evaluation (ID3-trained decision tree using HMM probabilities, ID3-trained decision tree using proportions, and heuristic-based approach) for each of the test sets.

in the heuristic-based original approach: in the case of the revisions pair $(7217, 7287)$, the good results in ranking end up producing a significantly smaller number of yes/no questions. However, for the remaining test sets, in which the ranking is not so good, even in the cases when it is better than the one obtained by the HMM, the mean number of questions is larger. Note that comparing both approaches in terms of ranking is difficult: while the heuristic-based approach uses a ranked list as the base for choosing the word forms to be posed to the user, the new approach uses a decision tree for this. In the case of the tree, the accumulation of probability in the correct candidate is notably more important than its position in a ranking, since this accumulation is what allows to reduce the number of questions to the user. This information nevertheless helps to understand the quality of the prediction of each strategy.

It is also important to analyse the impact of dictionary size in these results. Note that in the case of the decision-tree-based approaches, as the dictionary becomes larger, the number of yes/no questions necessary to determine the correct paradigm is also larger, although the growth rate is very slow. Similarly, the heuristic-based approach requires a larger number of questions as the dictionary size grows, although the heuristic strategy followed by the approach makes it more unstable and the differences between revisions larger. In the case of the approach using decision trees and HMM, the rising number of questions seems to be mitigated by the richer information available for disambiguating the training corpus.

Although a deeper analysis of the behaviour of the different approaches needs to be carried out, it can be concluded that decision-tree-based approaches are more stable and, in general, provide better results in terms of number of yes/no questions than the previous heuristic-based approach.

8 Conclusions and future work

In this work we have presented an approach that combines a hidden Markov model (HMM) and a binary decision tree in order to assist non-expert users in adding new words to monolingual dictionaries. This approach has been compared to the heuristic-based method proposed by Esplà-Gomis et al. (2011). The results have confirmed that the methods based on a decision tree are more stable and usually better than the original one. In addition, the comparison between the method using decision trees only and that combining decision trees and HMMs concluded that the number of queries asked in the second case is significantly lower. The Java code for the resulting system is available[17] under the free/open-source GNU General Public License.[18]

As regards future work, an extended evaluation including more pairs of languages and corpora would be necessary to confirm the results obtained here. It could be also interesting to try to improve the training corpus used, for example, by using a part-of-speech tagger to further reduce the number of compatible paradigms in L for each word form. Moreover, as pointed out in Section 5, a second part of the evaluation should still be performed to determine the feasibility of replacing the n-gram model proposed by Sánchez-Cartagena et al. (2012b) with the probabilities obtained with the HMM for choosing the correct paradigm among a set of equivalent ones.

Acknowledgements

This work has been partially funded by the Spanish Ministerio de Ciencia e Innovación through projects TIN2009-14009-C02-01 and TIN2012-32615, by 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

Bartusková, D. and R. Sedlácek. 2002. Tools for semi-automatic assignment of Czech nouns to dec-

[17] `https://apertium.svn.sourceforge.net/svnroot/apertium/branches/dictionary-enlargement`

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

Revision R_1	Revision R_2	Mean position of correct		Rate correct is first	
		HMM	Feasibility score	HMM	Feasibility score
7217	7287	1.47	**0.51**	70.31	**72.99**
11762	12415	**5.66**	10.45	**28.00**	8.36
17582	20212	1.87	1.72	**52.49**	40.88
27241	27627	7.11	**4.67**	39.73	**42.76**
34649	35985	6.66	**5.18**	45.57	45.57
36838	44118	**1.08**	3.51	**81.08**	70.52

Table 3: For the approach using decision trees and HMM and for the heuristic-based approach, mean position for each test set of the correct paradigm in the ranking of feasibility scores or probabilities and percentage of times that the correct candidate was the one with the highest score or probability.

lination patterns. In *Proceedings of the 5th International Conference on Text, Speech and Dialogue*, pages 159–164.

Baum, L. E. 1972. An inequality and associated maximization technique in statistical estimation for probabilistic functions of a Markov process. *Inequalities*, 3:1–8.

Bojar, O., C. Buck, C. Callison-Burch, C. Federmann, B. Haddow, P. Koehn, C. Monz, M. Post, R. Soricut, and L. Specia. 2013. Findings of the 2013 Workshop on Statistical Machine Translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 1–44.

Cutting, D., J. Kupiec, J. Pedersen, and P. Sibun. 1992. A practical part-of-speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*, pages 133–140.

Desai, S., J. Pawar, and P. Bhattacharyya. 2012. Automated paradigm selection for FSA based Konkani verb morphological analyzer. In *COLING (Demos)*, pages 103–110.

Détrez, G. and A. Ranta. 2012. Smart paradigms and the predictability and complexity of inflectional morphology. In *Proceedings of EACL*, pages 645–653.

Esplà-Gomis, M., V.M. Sánchez-Cartagena, and J.A. Pérez-Ortiz. 2011. Enlarging monolingual dictionaries for machine translation with active learning and non-expert users. In *Proceedings of RANLP*, pages 339–346.

Font-Llitjós, A. 2007. *Automatic improvement of machine translation systems*. Ph.D. thesis, Carnegie Mellon University.

Forcada, M.L., M. Ginestí-Rosell, J. Nordfalk, J. O'Regan, S. Ortiz-Rojas, J.A. Pérez-Ortiz, F. Sánchez-Martínez, G. Ramírez-Sánchez, and F.M. Tyers. 2011. Apertium: a free/open-source platform for rule-based machine translation. *Machine Translation*, 25(2):127–144.

McCreight, E.M. 1976. A space-economical suffix tree construction algorithm. *Journal of the Association for Computing Machinery*, 23:262–272, April.

McShane, M., S. Nirenburg, J. Cowie, and R. Zacharski. 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*. Ph.D. thesis, Carnegie Mellon University.

Quinlan, J. R. 1986. Induction of decision trees. *Machine Learning*, 1(1):81–106.

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

Šnajder, J. 2013. Models for Predicting the Inflectional Paradigm of Croatian Words. *Slovenščina 2.0: empirical, applied and interdisciplinary research*, 1(2):1–34.

Sánchez-Cartagena, V.M., M. Esplà-Gomis, and J.A. Pérez-Ortiz. 2012a. Source-language dictionaries help non-expert users to enlarge target-language dictionaries for machine translation. In *Proceedings of LREC*, pages 3422–3429.

Sánchez-Cartagena, V.M., M. Esplà-Gomis, F. Sánchez-Martínez, and J.A. Pérez-Ortiz. 2012b. Choosing the correct paradigm for unknown words in rule-based machine translation systems. In *Proceedings of the Third International Workshop on Free/Open-Source Rule-Based Machine Translation*, pages 27–39.

Walther, G. and L. Nicolas. 2011. Enriching morphological lexica through unsupervised derivational rule acquisition. In *Proceedings of the International Workshop on Lexical Resources*, Ljubljana, Slovenia.

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