Univ Access Inf Soc (2006) 4: 188–203
DOI 10.1007/s10209-005-0005-9

LONG PAPER

Nestor Garay-Vitoria · Julio Abascal

Text prediction systems: a survey

Published online: 8 December 2005

Abstract Text prediction is one of the most widely used techniques to enhance the communication rate in augmentative and alternative communication. Prediction systems are traditionally used by people with disabilities (e.g. people with motor and speech impairments). However, new applications, such as writing short text messages via mobile phones, have recently appeared. A vast amount of heterogeneous text prediction methods and techniques can be found in literature. Their heterogeneity makes it difficult to understand and compare them, in order to select the most convenient technique for a specific design. This paper presents a survey on text prediction techniques with the intention to provide a systematic view of this field. Prediction applications and related features, such as block size, dictionary structure, prediction method, user interface, etc., are examined. In addition, prediction measurement parameters and published results are compared. A large number of factors that may influence prediction results, including the acceptance of the system by the users, are reviewed, and their influence on the performance and usability of the system is discussed.

Keywords Text prediction · Word prediction · Anticipative interfaces · Augmentative and alternative communication · Communication speed enhancement

N. Garay-Vitoria (✉) · J. Abascal
Laboratory of Human-Computer Interaction
for Special Needs Informatika Fakultatea,
Euskal Herriko Unibertsitatea, Manuel Lardizabal, 1,
E-20018 Donostia, Spain
E-mail: nestor.garay@ehu.es
E-mail: julio.abascal@ehu.es
Tel.: +34-943-018000
Fax: +34-943-015590

1 Introduction

People with physical, perceptive and/or cognitive disabilities may have their communication ability reduced due to difficulties when using conventional interpersonal communication modes. Over the last few years, several computerised systems, usually based on augmentative and alternative communication (AAC) theories,[1] have been designed to assist people with disabilities for personal communication purposes. Using these systems, people with severe speech and motor disabilities[2] are able to overcome many communication limitations. For example, people with poor control of upper limbs may input text-using interfaces based on the selection of items, by means of a single key, from a set of elements sequentially scanned by the system. Scanning-and-selection input systems only require the handling of a residual voluntary controlled movement in any part of the body. Nevertheless, these communication devices are extremely slow, only allowing few words per minute. Alm et al. (1982) estimated that ten words per minute is the highest communication rate which can be obtained with these kind of systems[3] [6]. This may lead to embarrassing situations where disabled people cannot participate in normal conversations, and may therefore become socially excluded.

1.1 Human text prediction

Prior to the emergence of computer-based communicators, boards containing characters, syllables and words

[1] AAC tries to provide ways of communication to people who are not able to speak

[2] See the monograph [1] for a review of the disabilities that require AAC systems

[3] In a normal conversation, people can utter about 180–200 words/min, whereas a disabled person using a scanning-based input device can type only 2–10 words/min. This difference may cause practical problems for maintaining a conversation and psychological effects on the users [4]

were used for alternative communication. The interlocutor would look at the sequence of items pointed by the communication board user and reconstruct the sentence. The user frequently did not need to finish the word as the interlocutor was able to predict the remaining letters. To do this, the interlocutors used their knowledge of the structure of the language (morphologic and syntactic levels), and the context of the conversation (semantic and pragmatic levels). This prediction ability resulted in a considerable increase in the speed of communication, and a feeling of integration between both speakers. Therefore, the challenge for human interface designers is whether a computer is able to reproduce this behaviour, proposing sensible predictions based on the available information concerning the language and the context.

1.2 Users of prediction methods

As previously mentioned, predictors are frequently included in communication aids in order to increase the user's rate of communication. Therefore, these predictors may benefit people with severe motor and oral disabilities such as cerebral palsy, hemiplexia, etc., provided they are able to use them. According to the NCIP [54], users of prediction systems should have a certain phonetic spelling ability. They also need to distinguish words which look similar in prediction lists and must be able to flexibly shift their attention back and forth between the input device and the prediction list.

Even if prediction methods have primarily been designed to address the needs of people with disabilities, they can benefit anyone if they are adequately integrated into the user interface of an application. This is the case of the Reactive Keyboard [19, 35], and the T9 keyboard. The latter is commonly being used for composing SMS messages in mobile phones [62]. Text prediction can also be applied to other types of reduced keyboards (see, for e.g, [2] and [7]).

1.3 Diversity of prediction methods

Over the last few years, several research teams have reported many results on text prediction. They are based on diverse features of language and present important differences, including prediction algorithms, dictionary organisations, user interfaces, etc. A number of such systems are commercially available, while others have only reached the prototype phase. For these reasons, their performance can hardly be compared, as the measurements offered by authors are based on heterogeneous parameters, not always clearly described. Even if all these differences make comparison difficult, surveying known prediction methods can be useful to obtain an idea on the state-of-the-art in text prediction. For diverse reasons, commercially available systems are not included in this study. However, several authors, such as Gillette and Hoffman [33] or Heinisch and Hecht [38], published comparisons of commercial products. This survey is limited to methods which have been sufficiently described in published technical papers.

1.4 User interface issues

Prediction techniques use context information to anticipate what a person is going to write [23]. If the system is able to guess correctly, the number of keystrokes needed to write a sentence decreases and, apart from enhancing the speed of communication, the physical effort required to compose the messages is also reduced. In addition, the prediction software may also fix spelling mistakes, reorder sentences and more generally enhance the quality of the composed messages [52, 70].

Prediction systems can be seen as intelligent agents that assist users in composing texts. They capture user inputs to make their guesses and produce outputs which can be incorporated in the applications used to compose texts. As previously mentioned, they try to emulate user behaviour. Hence, the most advanced of them have learning features, are able to make inferences, adaptable and act independently. Moreover, in certain cases, they may converse with users, mainly to perform personal vocabulary adaptations.

Several user interface issues arise in systems using prediction techniques, and influence the efficiency of the predictive system. These issues are also discussed in this paper.

2 Analysis of prediction systems

In the context of assistive communication, a predictor is a system which tries to anticipate the next block of characters (letters, syllables, words, sentences, etc.) the user wants to express. In general, prediction is based on the previously produced blocks. The main aim of a predictor is to reduce effort and message-elaboration time. In order to reduce the effort, it is necessary to decrease the number of keystrokes needed for composing a message, whilst to reduce the needed time, the number of characters incorporated into the text by means of a single prediction should be larger than the number of characters written by a single selection.

There are two main concepts that have to be taken into account when analysing communication systems, *articulateness* and *fluency* [37]. An input system has high *articulateness* if there is always at least one option fitting the user needs which can be selected among all offered proposals. That is to say, the character(s) the user wants to write is (are) always among those offered by the system. *Fluency*, which is directly connected to the communication rate, is the number of characters produced on a time unit.

Articulateness and *fluency* are opposite concepts, as an increase of one usually causes a decrease in the other. A system which offers only the alphabet to the user is an example of a very articulate but poorly fluent system, as when using it, a choice can always be selected, but the communication rate obtained is low when the user has several motor impairments. A very fluent but poorly articulate system is the one which offers several common words or sentences. If one of the options is selected, the communication rate is greater than when having to compose words letter by letter; however, the required option may not be in the selection set.

The aim of a communication-aid device is to maximise and balance both *articulateness* and *fluency*. When trying to obtain a high level of *articulateness*, all the possible characters are offered to users. In the case of *fluency*, in addition to the alphabet characters, various options longer than one character should be proposed. If these options are frequently selected, *fluency* increases while *articulateness* remains of a high standard, as at least one of the shown options may be selected each time.

The next sections describe a number of important factors related to prediction systems, including block size, dictionary structure, prediction method, influence of the language, the user interface, prediction measurement, usability-related issues, adaptability and special features. A number of such factors are related to the internal functioning, whilst others are affected by characteristics of the users who are going to make use of the prediction systems.

3 Text block size

The selection of the size of predictable blocks is a very important factor, as keystroke rate and time saving depend on the size of the anticipated block. Therefore, the possible savings are a function of the block size and the hit ratio of the predictor [30]. In this paper, only blocks composed of alphabetic characters are considered (other symbols and pictograms are not taken into account). Focus is therefore on text prediction.

For pure text, the minimum block size is the character. In each language, each character has a different frequency of appearance. A first approach would propose all the letters sorted by frequency (such as in Table 1), without taking the preceding composed text into account (that is, with zero memory). In this way, the proposals are always the same, independent of the previous characters. It is evident that this method does not produce any keystroke saving as it requires the same number of keystrokes to accept the proposals and to write these characters in the case of success. Nevertheless, the information of the character frequencies is useful in the scan-based interfaces to minimise the average time access [50, 65]. Hence, for a speedier access the scanning options are distributed depending on their frequencies.

As predicting just one character produces very poor results, higher blocks must be treated. In order to make use of the available statistic information coming from the precedent text, *n-grams* (that is, blocks of *n* characters) are frequently used. If we have a table containing *n*-grams frequencies, it is possible to guess the *n*-th character after a block of *n*-1 characters [19, 35, 50]. The hit ratio and keystroke saving increases with *n*, until an upper limitis dependent on the language. The information related to *n*-grams is used to display dynamic scanning matrixes where the most probable (with respect to the last selected characters) are placed near to the top-left place in the matrix. Lesher et al. [50] carried out a study related to this type of prediction and compared it with a word prediction method.

Traditionally, the maximum length of the *n*-grams is four (tetra-grams), mainly due to computational and storage limitations [50]. However, with reference to classical Shannon's [59] studies, it has been estimated that adequate predictions only require the knowledge of few characters and there are no significant improvements in the achieved results when knowing more than eight previous characters.

Special cases are *k-grams*, which are *n*-grams whose first character is the start of a word. *N*-grams are independent of their position within a word, whereas *k*-grams take advantage of the positioning [50].

Other blocks to take into account are syllables and morphemes. As they are "natural" parts of words, they have interesting linguistic properties which are very useful for prediction purposes. Though the number of characters in a syllable is variable (let us say ***s*** characters per syllable in average), the number of syllables that are used in a given language is smaller than the number of *n*-grams that can be calculated. Therefore, the number of syllables to store is smaller, hence increasing the probability of guessing them.

Even if syllables and *n*-grams are highly predictable, the users experience a certain amount of difficulty in order to identify and integrate them into text. Nevertheless, several authors (such as Lesher et al. [50]) proposed systems which try to guess a combination of single characters, *n*-grams and syllables. Although savings obtained with the smaller blocks are low, they are kept into the predictable set to maintain the consistency of the system. The reason is that in this kind of systems, the user tends to concentrate on the prediction zone, where all possibilities (even small blocks) appear.

On the other hand, the set of morphemes in a language is composed of roots of words and possible affixes (prefixes, infixes and suffixes). In this case, it is necessary that both the system and the user are able to clearly distinguish what a morpheme is. These blocks are very interesting in highly inflected languages (as in German, Swedish, Basque, etc.), as these languages create a large number of different forms combining diverse morphemes, and it would be almost impossible to create a lexicon including all these possible forms. Sect. 6 discusses this issue in greater detail.

In the perspective introduced above, the word is considered as a very adequate block due to its linguistic properties and average length. Word prediction systems

Table 1 Character frequencies for different languages (values are only approximate as each author uses different criteria to measure relative frequencies)

English[a]		Spanish[b]		German[c]		French[d]		Basque[e]	
Ch.	Fr. (%)	Ch.	Fr. (%)	Ch.	Fr. (%)	Ch.	Fr. (%)	Ch.	Fr. (%)
E	12.77	E	16.78	E	16.69	E	16.26	A	15.38
T	8.55	A	11.96	N	9.91	S	7.80	E	13.04
O	8.07	O	8.69	I	7.81	A	7.77	I	8.76
A	7.78	L	8.37	S	6.77	I	7.03	N	8.39
N	6.86	S	7.88	T	6.74	T	6.70	R	7.92
I	6.67	N	7.01	R	6.54	N	6,67	T	7.36
R	6.51	D	6.87	A	6,51	R	6.22	O	5.35
S	6.22	R	4.94	D	5.41	U	5.77	Z	5.24
H	6.95	U	4.80	H	4.06	L	5.44	U	4.92
D	4.02	I	4.15	U	3.70	O	5.15	K	4.85
L	3.72	T	3.31	G	3.65	D	3.54	D	3.03
U	3.08	C	2.92	M	3.01	C	3.03	B	2.99
C	2.96	P	2.78	C	2.84	M	2.74	L	2.65
M	2.88	M	2.12	L	2.83	P	2.70	G	2.60
P	2.23	Y	1.54	B	2.57	V	1.40	S	2.53
F	1.97	Q	1.53	O	2.29	Q	1.05	H	1.88
Y	1.96	B	0.92	F	2.04	F	1.02	M	1.06
W	1.76	H	0.89	K	1.88	G	0.97	P	0.91
G	1.74	G	0.73	W	1.40	B	0,89	J	0.37
B	1.41	F	0.52	V	1.07	H	0,81	X	0.35
V	1.12	V	0.39	Z	1.00	J	0.48	F	0.23
K	0.74	J	0.30	P	0.94	X	0.41	C	0.09
J	0.51	Ñ	0.29	J	0.19	Y	0.27	V	0.04
X	0.27	Z	0.15	Q	0.06	Z	0.12	Ñ	0.02
Z	0.17	X	0.06	Y	0.03	Ç	0.08	Y	0.02
Q	0.08	K	0.01	X	0.02	K	0.05	W	0.01
		W	0.01			W	0.03	Q	0.01

[a] Cronos Word. A Challengers Handbook. http://www.caesum.com/handbook/tech.htm
[b] El rincón de Quevedo. http://rinconquevedo.iespana.es/rinconquevedo/Criptografia/frecuencia.htm
[c] Santa Cruz Public Libraries http://www.santacruzpl.org/readyref/files/g-l/ltfrqger.shtml
[d] Lexique 2. Fréquence des Lettres. http://www.lexique.org/listes/liste_lettres.php
[e] From Gonzalez-Abascal [34]

try to guess the word the user is trying to write, taking previously written characters into account. This strategy usually produces satisfactory hit ratios (as words are frequently highly predictable) and an important keystroke saving. In addition, the user's cognitive effort is low as words are easily distinguishable and their separation is clear (in opposition to *n*-grams, syllables and morphemes).

Another option is to try blocks longer than one word. In this case, the system tries to anticipate the following words, taking the last composed words(s) into account. In case of correct guessing, keystroke and time saving would be very high; however, the hit ratio for more than one word is very low as the frequencies of sequences of words are extremely low, except in a few common sentences, as in filler remarks, greetings and farewells [6, 25]. A valid example is the conversation helped by automatic talk (CHAT) system implemented by Alm et al. [6]. However, its utility for general topic conversation is difficult to estimate.

To sum up, among all possible blocks, the word is usually selected due to its optimum relationship between hit ratio, keystroke saving and cognitive cost on the part of users. Furthermore, the addition of morphologic, syntactic and semantic information may enhance the hit ratio and keystroke saving. Therefore, the word is the most widely used predictive block [17, 21, 26, 38, 39, 40, 45, 60, 63, 67]. Nevertheless, there are also predictors which work with *n*-grams [35, 50] and morphemes [11, 27, 43].

4 Dictionary structure

Prediction based on characters or *n*-grams only need a table to store data on the blocks and their frequencies. On the other hand, word prediction systems store the information they require in dictionaries or lexicons [28]. Predictors which treat units longer than one word have larger storage requirements. With each sentence, they usually have to store information on the context, block type, frequency, domain of conversation, etc. [6, 25].

To select the lexicon structure, the following factors must be taken into account [60]: speed of operation (either access or modification), facility of modification, price of storage, storage simplicity and backup charge/discharge. The possibility of compressing files must also be taken into account.

4.1 Dictionary organisation

The organisation of the lexicon may be a linear list where items (that is, words and their related information) are stored sequentially [60]. The order in which items are stored may be ascendant or descendant, either alphabetically or by frequencies [28]. This organisation is very simple: it is easy to carry out searches and modifications. However, the main disadvantage is that it is very slow, as access is mainly sequential. Therefore, in large lexicons, obtaining access to the last items requires a long time.

In contrast to the linear list organisation, a number of lexicons have a tree structure. In these dictionaries, searches are faster; however, the structure and the management of the lexicon are more complex, and adding a new element may cause changes in the entire lexicon structure (see Fig. 1).

4.2 Number of dictionaries

It is possible to have only one lexicon which stores all the items or to have diverse lexicons to distribute the items. The first option may provide a longer access time but a simpler management. The second option may provide faster access but complex management due to the number of dictionaries which have to be treated. Management of multiple small lexicons may be especially complex, and the treatment of all of them may become slower than in the case of a single large lexicon.

4.3 Adaptation of the dictionaries

The system is usually provided with an initial lexicon common to all users. This lexicon may have a standard set of words with their frequencies. In this case, the system can produce statistical predictions without an initial setup. Adaptive dictionaries must be provided with methods to update the frequency of use of the words, as each person uses different words, and with different frequency. This method allows for increasing of frequency of commonly used words (firstly proposed), and decreasing the frequency of words more rarely used.

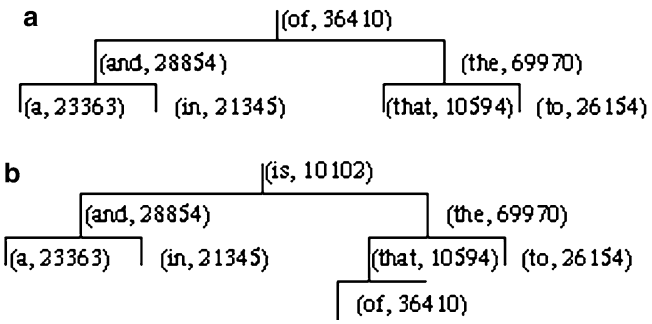


Fig. 1 Example of tree structured-lexicon (**a**) and the possible reorganization required after including word "is" as the root of the tree (**b**). Items are composed of a word and its absolute frequency between *brackets*

It is also possible to create an initial lexicon by gathering the words frequently used from a sample of representative users. A number of studies estimate this set of words for certain languages. For example, Beukelman et al. [12] showed how to create an initial lexicon for English speaking disabled people. Apart from these studies, the frequencies of the words appearing in standard literary texts or in the written press are frequently used to create the initial lexicon.

Once created, the lexicon can be adapted to the user's vocabulary. The user's personal satisfaction with the system increases as words coming from personal vocabulary are promoted and, therefore, better results may be achieved. Consequently, dictionary adaptation is a very interesting feature and most systems include it, for e.g, the systems presented by Swiffin et al. [60] and Venkatagiri [64].

However, adaptation can create new problems, some of which are:

- A negative effect of the adaptation may occur when proposals appear in different places or order (as their frequency has changed). Koester and Levine [47] believe that this feature is not very much appreciated by users who prefer a static presentation of the proposals.
- When adaptation is only incremental, that is, new words can be added to the lexicon and words which are not used cannot be eliminated, the size of the dictionary increases indefinitely. Moreover, misspelled words may be included in the dictionary and never disappear (even if they are no longer used). In addition, the time required for searching all possible words also increases. Therefore, there is a need to limit the dimensions of the lexicon in order to limit the time necessary for searching best proposals. "Garbage collection" procedures are also proposed for these systems.
- A number of systems classify words within sets of thematic dictionaries [41]. In this case, in order to include a new word, it is necessary to define its semantic tag.
- If new words are included to a system that uses morpho-syntactic information, it is necessary to provide this information (and sometimes an initial frequency) to be associated to the word. The ideal solution would be a system which automatically includes this information in the lexicon. As morpho-syntactic information is not always evident, the system may include erroneous categories which would make the subsequent predictions difficult. Other strategies allow the system to ask the user about the characteristics of the new words when they appear. If the user does not know the morpho-syntactic characteristics of a word, similar problems arise. In addition, this strategy decreases the message composition rate. To avoid this, new words are included with a void category and afterwards the user is required to complete the required information.

To summarise, adaptable lexicons have proven to be very useful as they can obtain better results in terms of keystroke saving. Nevertheless, as they present important problems, intelligent decisions have to be taken in order to obtain the best results, taking the user's preferences and abilities into account.

5 Prediction method

Diverse strategies may be used to make predictions. This section focuses on word prediction methods. However, these methods may also be applied, with subtle variations, to other text predictors.

5.1 Prediction using frequencies

The simplest predictors take the words and their frequencies[4] into account [23, 37, 41, 60, 64]. When the user has written the beginning of a word, the system offers the most probable words beginning with the same character(s). The user may then accept one of the proposals or may continue writing the word (if the desired word is not among the options given by the system). The system may automatically adapt to the user's lexicon by simply updating the frequencies of the used words. If a new word is added to the system, an initial frequency is assigned.

In order to enhance the results of this approach, it is possible to record the word used last time, that is, the *recency* of use. In this case, the system offers the words most recently used, selected among the most probable words which begin with the character(s) written by the user. A *recency* field is stored in the dictionary with each word along with its frequency. Although the results obtained with this method are better than the ones based on frequency alone [60], the method requires storage of more information and increases computational complexity.

5.2 Prediction using word probability tables

Another possibility is to consider the probability of appearance of each word after the one previously composed. A two dimensional table is built, where the conditional probability of the word W_j after the word W_i is stored. Therefore, if the system has N words, there are N^2 entries in this table, most of them zero or nearly zero. By using this strategy, the system may offer predictions before the user starts writing the initial character of a word. These results may be enhanced if recency is also taken into account. The main problem with the table-based implementation of this method is the difficulty of adaptation to the user's vocabulary when the dimensions of the table are fixed. Therefore, this approach is normally restricted. For example, several authors, (e.g., Hunnicut [41]), only use this approach with the most probable word-pairs.

5.3 Syntactic prediction using probability tables

This approach takes the syntactic information inherent to natural languages into account. For this purpose, two statistical data types are used, the probability of appearance of each word and the relative probability of appearance of every syntactic category after each syntactic category. These systems offer the words with the most probable categories at the current position of the sentence. The results are usually better than the ones obtained with the previously shown strategies.

A word in the dictionary is associated within its syntactic category and its frequency of appearance. For words with ambiguous categories, diverse strategies are discussed by Garay in [26]. The most common strategies are either to create a new category related to each possible reading of the word, or to associate a list of categories to each word.

A two-dimensional table stores the probabilities of appearance of the categories after each category. Its dimensions are fixed before the predictive system is built. This table is much smaller than the one in the previously presented approach and the number of probabilities which are nearly zero is also lower. Adaptation of these systems is performed by updating the probabilities of the table and the frequencies in the lexicon. If new words are added, some of the strategies discussed in Sect. 4.3 have to be applied. Several systems implementing this strategy have been developed [23, 61].

5.4 Syntactic prediction using grammars

Sentences are analysed using grammars and by applying Natural Language Processing techniques in order to obtain the categories which have the highest probability of appearance. Methods for analysing the sentences may be top-down [63] or bottom-up [24, 26]. Each natural language has a set of syntactic rules. These rules usually have the following structure:

$$\text{LEFT} \leftarrow [\text{RIGHT}]+ . \qquad (1)$$

This means that the category in the left part of the rule may be decomposed in the sequence of categories in the right part, in the order in which they appear. At least one category has to appear on the right of the arrow and all the categories are defined in the system. For example, if NP represents a Noun Phrase, PP a Prepositional Phrase, Noun a noun and Prep a preposition, the rules:

[4] Standard frequencies from dictionaries are usually taken. If it is possible, it is much better to use the frequencies of the words used by each user. Additionally, there are studies related to the frequencies of the words for different populations, for instance the previously mentioned [12]

$$\mathrm{NP} \leftarrow \mathrm{Noun\ PP}. \tag{2}$$

$$\mathrm{PP} \leftarrow \mathrm{Prep\ NP}. \tag{3}$$

may be expanded producing:

$$\mathrm{NP} \leftarrow \mathrm{Noun\ Prep\ NP}. \tag{4}$$

Therefore, recursive application of the rules is possible. In addition, it is possible to define a number of morphological agreement constraints (e.g., grammatical number) among the categories on the right of a rule. In this way, the predictor can offer proposals taking appropriate morphological characteristics into account.

The used dictionary is very similar to the one in the previous approach. However, in order to enforce morphological agreement, it requires the addition of morphological information. These systems have a higher computational complexity than the previous ones, mainly due to the fact that they take the entire beginning of the sentence into account (while the previous systems take, at the most, the last entirely composed word). These types of systems may be adapted by updating the word probabilities and weights of the syntactic rules [42, 67]. To add new words to the dictionary, the solutions mentioned in Sect. 4.3 may also be used.

5.5 Semantic prediction

These methods are not commonly used, as the results they achieve are very similar to syntactic approaches with a much higher computational complexity. The easiest way to make this type of prediction is to semantically analyse sentences as they are being composed. Each word has an associated semantic category (or a set of semantic categories), similar to the syntactic categories of the previous last two approaches. The remaining characteristics (working method, complexity, dictionary structure, adaptations, etc.) are very similar to the syntactic approach using grammars. However, the main difficulty is to add a semantic category to each word. Several authors proposed categorisation ''by hand'' [42]. There are several automatic allocation methods which use semantic information [24, 41]. However, they are quite complex and it is difficult to implement them in a real-time system. The large amount of time they require to make predictions is the main reason why they are mostly academic approaches.

6 The influence of language in predictions

Most of the work related to prediction concerns low-inflected languages such as English. This paper will also consider highly inflected languages such as German, Swedish, Finnish, Basque, etc. A language is inflected when it is possible to produce morphological forms starting from a root or lemma. The root is the part representing the main idea of the word and remains unchanged in every form of the same semantic family. The degree of inflection of a language may vary from very high (e.g., Basque[5]), to moderate (e.g., Spanish, French), to low (e.g., English) [28, 29].

Morphological forms mainly express the number (and, in some cases, the gender). For instance, the number may be singular or plural (*house*/*houses*, *spy*/*spies*, etc.). In certain languages (German, Spanish, etc.), the gender may be masculine, feminine or neuter.

When there are only few variations of a word, it is possible to store all of them in the dictionary. However, as highly inflected languages produce many forms, it may be difficult to store all of them. This is the main reason for the search of new prediction methods in languages with a wide use of prefixes, infixes and suffixes. Related work is reported in [10, 11, 14, 27, 29, 43].

6.1 Morpho-syntactic correctness

An interesting feature of word prediction is the possibility of showing the proposals in the most appropriate morpho-syntactic form. It is also possible to automatically correct the resulting text after accepting a proposal in order to obtain the most appropriate form [70].

When the system does not have any type of correction, the user has to manually adapt the suffixes of the accepted proposal. For example, if the user wishes to write ''houses'' and the system only proposes ''house'' he or she must add an ''s''. This occurs as the context of the sentence is not taken into account, as in the case of prediction using frequencies (see Sect. 5.1 in this paper). For languages which are not highly inflected, this may be a perfectly valid approach. However, in highly inflected languages, the number of possible suffixes is very high and a great physical (and possibly cognitive) effort is needed to add the appropriate suffix to the base form.

In order to solve this problem, several authors propose dividing the prediction of a word into *root* and *suffix* [11, 27]. In this way, the root is firstly composed and then a list of suffixes is proposed to create the appropriate form. In general, a two-step process is needed to entirely compose a word (there may be a few exceptions) and post-processing may be necessary (for example, to generate irregular cases starting from a root and a suffix). Therefore, there is a need to have at least two lexicons, one for roots and one for suffixes [29].

[5] In the case of the Basque language, starting from a given root, 62 basic inflections may be obtained. Suffixes may be recursively concatenated (as it is an agglutinated language) increasing the number of possible inflections. With a two-level recursion, it has been estimated that a noun may reach 458,683 variations [3]. Prefixes and infixes are also possible in Basque but it has been found that their frequencies are not very significant comparing to suffixes [29]

There may also be a number of cases where roots and suffixes may be concatenated to provide entire words as proposals [28].

An interesting option to avoid this extra effort is the possibility of automatically adapting accepted proposals to the correct morpho-syntactic form that best matches the current sentence.

A third possibility for solving this problem is to make systems able to directly present the proposals in the most adequate form. This is only used in systems working with low-inflected languages, as it can be hardly achieved in inflected languages. This approach is the one which requires the highest computational effort and the least effort on the user's part. Nevertheless, if the proposed word form is not what the user wants to write, it still must be manually corrected.

The amount of effort required from the user (with impact on both fatigue and lower composition rate) depends on the approach used to solve this problem.

7 User interface issues

The main goal of a predictive system is to propose a set of words that include the word required by the user. In case the predictive system is able to guess the desired word, the user selects it, and the system incorporates it in the text being edited. Due to the interactive nature of this process, the above sequence of tasks must take place in a limited period of time in order to be useful for the user. However, even when the system is able to guess the word, it may also occur that the savings obtained are not worthy. That is, the user is not able to obtain any advantage from the prediction made. The reason for this can be found in the way in which proposals are presented to the user, that sometimes impede or make it difficult for the user to take advantage of them. Furthermore, the cognitive cost associated to the use of the predictor also has an influence on the usability of the system. Both factors are related to the system's user interface.

In the following sub-sections, some of the user interface characteristics that contribute to prediction efficiency are discussed [28, 31].

7.1 User interaction

According to [35], the main user-system interaction methods in text prediction systems are:

- The system gives a prediction list and the user has to select the desired one. This procedure has the disadvantage that a prediction always has to be explicitly selected, with the associated effort.
- The system directly introduces the predicted word in the message. If the word is appropriate, the user continues writing the text. If the word is not the desired one, the user has to reject it by means of a specific action (also called *explicit rejection protocol*). The advantage is that the user has to act only if an erroneous prediction is made. This requires less effort, although it has the disadvantage that if the user skips one of the predictions, the produced text is different from the expected one.
- The system includes the predictions in the selection set (ordered by probability). Following the general procedure to insert text, the user either selects one of the proposals or continues inputting letters if none of the predictions is adequate (also called *implicit rejection protocol*). This method does not increase the amount of physical effort needed by the user to write, but it may require a higher level of concentration due to the need to read the proposals.

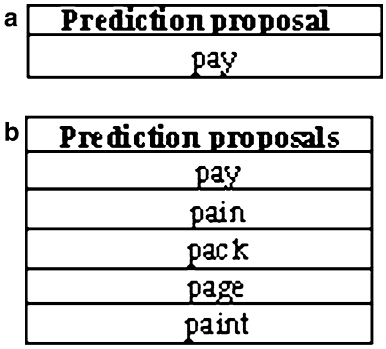
Therefore, the decision about where and when to locate the proposals may be determinant for the acceptability of a prediction system.

7.2 Number of proposals

One of the most important usability factors is the number of proposals provided by the system. Most papers related to the prediction mention that the larger the number of proposals, the better the prediction results are (e.g. [17, 26, 47, 60]). However, a high number of proposals largely increase the time and cognitive effort required to select a proposal. Koester and Levine [47] calculated that each additional word increases the searching time across a proposal list by 150 ms. The problem of reading the options is especially relevant when the user has severe motor impairments and the keyboard and screen are in different visual planes. The required time and effort to move the head in order to read the proposals and then to move it to select an option in the keyboard (estimated by Koester and Levine in nearly 0.8 s for each option) may significantly slow the message composition speed [46]. This is the reason why various authors propose making the visual planes of the keyboard and the screen as close to each other as possible [56]. Logically, there is a conflict between the number of proposals and the achieved message composition rate. Even if several authors designed and tested prediction systems with a large number of proposals (up to 20 in [17]), no more than five to seven proposals are usually offered. The reason is that many authors (such as [38, 44, 60]) have estimated that this is the number a user may perceive at a glance. For instance, Fig. 2 [a] shows an interface which offers a proposal while the interface in Fig. 2 [b] suggests five.

7.3 Distribution of the options

Apart from the number of proposals, the way the proposals are presented is also very relevant in order to



Prediction proposal
pay

Prediction proposals
pay
pain
pack
page
paint

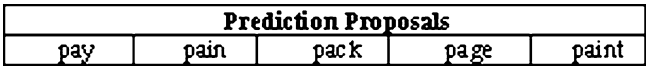
Fig. 2 Single (**a**) versus multiple (**b**) choices for prediction

speed up the selection process. In general, showing the proposals as vertical word lists (Fig. 2 [b]) is better accepted than the horizontal presentation (Fig. 3), as the required effort to see and process in vertical lists is lower [44, 56, 60].

Furthermore, the order of the proposals is important for efficiency. The most frequent rule is to sort them by their probability (Fig. 2 [b]). When the number of proposals is high, the use of alphabetically ordered lists may lead to an easier selection (as in Fig. 4) and to increased speed. A number of users may prefer the proposals always located in the same position, as it is easier to memorise their allocation and automate their use [47].

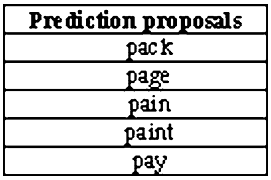
7.4 Layout of the selection set

In scan-based interfaces, the layout of the proposals must be carefully analysed. Triangular matrixes (Fig. 5) give the best average access time [8]; however, they are difficult to present on a screen. There is also the possibility (known as grouped access [50]) of distributing the options between more than one matrix, as depicted in Fig. 6. In both Figures, "n^{th} option" (or "n^{th}") are the proposals given by the predictor (usually ordered by probabilities), "blank" is the space character, and the remaining characters are letters and punctuation signs. For instance, in Fig. 5, scanning may start highlighting rows top-down. When a row is selected, its elements are scanned from left to right until one is selected. In Fig. 6, the block is first selected (either the list of predictor proposals on the left or a matrix with the characters on the right). If the left block is selected, the desired word is directly incorporated in the text with a new keystroke needed to choose it. On the other hand, if the right block is selected, two more keystrokes are needed (one for the row and another for the column) to choose the next character the user wishes to write.



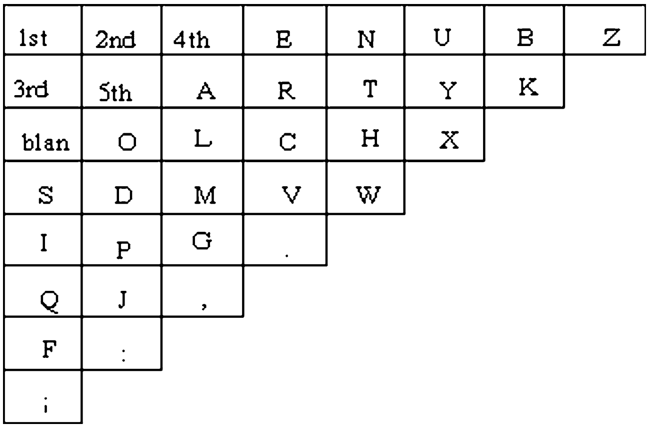
Prediction Proposals				
pay	pain	pack	page	paint

Fig. 3 Horizontal presentation for prediction. Proposals are shown considering their probabilities



Prediction proposals
pack
page
pain
paint
pay

Fig. 4 Alphabetically ordered vertical list of proposals



1st	2nd	4th	E	N	U	B	Z
3rd	5th	A	R	T	Y	K	
blan	O	L	C	H	X		
S	D	M	V	W			
I	P	G	.				
Q	J	,					
F	:						
;							

Fig. 5 Example of a triangular matrix

7.5 Required time for prediction

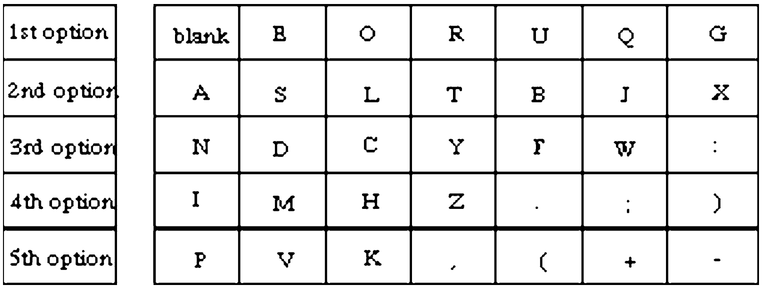
Highly complex prediction algorithms may lead to inefficient systems. Large lexicons may also increase search time. There is a need to find a balance, for each individual user, between algorithm complexity, dictionary size and response time.

8 Measurement of prediction results

The hit ratio is the probability of guessing in prediction and it may be expressed as the quotient between the number of times the words are guessed and the number of written words (that is, it is the probability of guessing the word before completely writing it):

$$\text{hit ratio} = \left(\begin{matrix}\text{no. of times the} \\ \text{word is guessed}\end{matrix}\right) \Big/ \left(\begin{matrix}\text{no. of written} \\ \text{words}\end{matrix}\right). \tag{5}$$

In general, a predictor is considered to be adequate if its hit ratio is high, as the required number of selections decreases (as seen in [17, 23, 41, 50, 60]). A high hit ratio may be achieved by giving the appropriate choosing criteria to the predictor. However, the hit ratio is not enough to evaluate the quality of a predictor, as it is possible for the valid guesses to occur at the end of the word, with very low keystroke saving. This is the reason for including more parameters to measure this feature. For instance, keystroke savings are defined as:



1st option	blank	B	O	R	U	Q	G
2nd option	A	S	L	T	B	J	X
3rd option	N	D	C	Y	F	W	:
4th option	I	M	H	Z	.	;	)
5th option	P	V	K	,	(	+	-

Fig. 6 Example of the grouped access for a scan-based interface

$$\text{keystroke saving} = 1 - \left(\begin{matrix}\text{no. of written}\\ \text{keystrokes}\end{matrix}\right) \Big/ \left(\begin{matrix}\text{total no. of}\\ \text{keystrokes}\end{matrix}\right). \tag{6}$$

In direct selection, each keystroke produces a character. Therefore, it is easy to obtain character saving from this equation.

Time saving is also defined in a similar way:

$$\text{time saving} = 1 \text{ -- } (\text{used time})/(\text{necessary total time}) . \tag{7}$$

This last measurement is more difficult to evaluate, as there are various factors that depend on the user's characteristics. For this reason, time saving is usually estimated as a factor proportional to keystroke saving.

All these measurements assume that only one proposal is offered each time. It is evident that the hit ratio improves when presenting a number of options instead of only one. However, after a given number of options, the hit ratio grows very slowly (its growth presents an asymptote) [17, 28]. Moreover, by incrementing the number of options, the probability of guessing sooner grows. Nevertheless, the required time to select among them also increases, as mentioned in Sect. 7.2. Therefore, it is not very useful to increase the number of options as the delay in selecting the appropriate option is not as good as the advantages of the prediction.

There are other methods to evaluate the quality of the predictors. For instance, Greenberg et al. [35] present the potential savings as the difference between the cost of using predictions and the cost of entering standard events, as presented in Table 2. In both costs, the time required to decide and to execute its relative actions is included. Even-Zohar and Roth [21] use the word error rate and the errors in the focus of attention (that is, when the correct word and its category are not guessed) to measure prediction performance.

8.1 Expressing the results

Generally, prediction methods found in the literature express their results in terms of the savings they achieve. For example, most predictors have been evaluated by calculating the obtained keystroke saving [23, 24, 41, 42, 60, 61]. Other studies calculate the message composition time saved, which is directly associated with keystroke saving [17, 45]. The hit ratio may also be reported [28], as previously noted.

A rigorous comparison among the different methods appearing in the literature would require a standard workbench to establish a corpus from where to obtain the dictionaries, the trial texts to write and the type of measurements to obtain [28]. As this workbench does currently not exist, the results presented by several authors are very heterogeneous and practically impossible to reproduce (as the description of the used algorithms is usually superficial). Therefore, comparisons among them are purely estimative and do not have evidential value.

8.2 Evaluation methods

There are two main methods to evaluate the presented systems in the literature related to prediction. The first, *emulation*, feeds the predictor with several trial texts, emulating human behaviour. The second method, *human testing*, employs a sample of real users who try to write specific texts.

Emulation always produces the best possible results, in terms of time saving, as the programme "types" at the highest possible speed. Therefore, this method has the disadvantage that it does not take into account human features, such as fatigue or error making. Nevertheless, this method is relatively objective and it is useful to take a number of design decisions. Many authors show the results when using evaluation methods [15, 23, 39, 41, 60, 68].

Table 2 Potential savings using prediction

Savings = cost of using predictions - cost of entering standard events
Cost of using predictions = visual scan time to search predictions + cognitive time to decide on prediction + physical movement time to select prediction
Cost of entering standard events = cognitive time required to formulate next event + physical movement required to enter next event

On the other hand, empirical evaluation with users allows obtaining more reliable data with regard to the message composition speed as the test includes human factors such as error rate, fatigue, learning time, users subjective satisfaction, etc. Moreover, it is possible to measure how the system is used (for example, if the user reads all the proposals the system provides before making a selection), whether the system and its interface address user requirements (e.g., if it is easy to learn and use, if a faster message composition is achieved, if better orthography is achieved, and so on). Data related to the user's preferences may also be obtained. Therefore, several authors prefer to perform tests with real users [15, 45, 55, 64].

The perspective adopted in this paper is that the second method is the best in order to obtain a validation of the design in real conditions. However, it presents the difficulty of identifying a set of people who may represent the entire population of possible users of the system. Moreover, the confidentiality of the results may also be a problem [18] and there are authors who suggest that this type of study may be very limited and confined to individual case studies [20]. This is the reason why, in the design phase (testing and selecting the different characteristics of the system), predictors are frequently evaluated by simulations in order to compare their characteristics in an objective way.

8.3 Prediction results

Not all the authors present the obtained results with the methods they use. In addition, as previously indicated, the lack of a standard workbench makes it impossible to compare each approach with the others. This section presents a number of relevant results found in the literature.

Firstly, the results achieved with the predictive adaptative lexicon (PAL) system and the SYNTAX PAL (as shown in [60, 61]) are presented. With an initially empty lexicon, keystroke saving vary between 29 and 41% by using only statistical information. With pre-built lexicons, results may vary between 50 and 55%. Adding certain syntactical information, the results are enhanced between 0.5 and 2.0% with regard to the purely statistical version. Taking only syntactical information into account (without statistical data), the authors note that the results are enhanced between 4.3 and 6.4%. In all the studies, the number of proposed words is one, five or ten. From five proposals onwards, there is no significant enhancement of the results. The authors also state that they have obtained a 69.48% for keystroke saving, which is very near the theoretical maximum.

With the CHAT system [4, 5, 6], the authors suggest that the achieved communication speed is between 12 and 85 words/min, still far from the 150–200 words/min of natural oral conversations, but better than the 2–10 words/min achieved with the use of typical communication devices.

In the studies related to the PROFET programme, a statistical approach is investigated in comparison to the addition of syntactic or semantic information [41, 42]. The authors found that the addition of syntactic information enhances the results of a purely statistic approach between 2.6 and 5.1% (taking one to six proposals into account). Surprisingly, they also found that the addition of semantic information does not enhance these results. In the studies devoted to PROFET II [15], the grammatical monograms have been found to enhance the results of the statistical approach by 3.1%. Grammatical bigrams enhance the monogram approach by 7.4%, and a tri-grammatical approach enhances the bi-grammatical one by 1%. The new version of the system achieves a 47% in keystroke saving in comparison with the 35% achieved by the older version. With a larger lexicon, keystroke saving is near 50%.

A number of different predictive programmes are evaluated using the same trial tests in [39]. These systems are EZ Keys, predictive linguistic programme (PLP), Write 100, words strategy and generic encoding technology (GET). Several tests with 20,500 word-length texts are composed by scholars without disabilities. Keystroke saving vary between 31% (GET) and 45% (EZ Keys). The vocabulary covered by these methods is also mentioned, and varies from the 45% in Write 100 to the 91% in EZ Keys.

The results of the Messenger programme ([64]) with an initial lexicon of 903 words and showing up to 15 proposals for the user are the following: keystroke saving is 49.1%, 88% of the words are predicted and the time needed to compose the messages is greater when using prediction than without (in a proportion of 1.057 to 1.0).

The studies carried out by Lesher et al. [50] related to prediction and applied to scanning-based interfaces, show that using *n*-grams and *k*-grams, keystroke saving is up to 40.5%, while word prediction achieves a saving up to 38.8% when presenting seven proposals. The authors conclude that it is easier to guess one out of 26 characters than one out of 50,000 words.

The WindMill system achieves up to 56.33% in keystroke saving taking both statistical and syntactic information into account [68], and nearly 100% in hit ratio after entering the initial three letters of a word.

In the versatile interpretation of text input by persons with impairments (VITIPI) system presented by Boissiere et al. [13], the achieved keystroke saving is up to 29.3% for French (with a 5,930 word lexicon), up to 35.3% for English (with 2,566 words in the lexicon) and up to 44.3% for German (with 5,835 words). The system is also claimed to increment the communication speed and improve the quality of the composed text with respect to HandiWord, which is one of the systems comparable to VITIPI.

The Predice and PredictAbility systems are evaluated by using a number of multilingual trial texts (in English, French, German, Italian and Spanish) published by the European Union [57]. With five proposals, keystroke saving obtained by PredictAbility vary between 30 and 37%. They suggest that these results may be enhanced (for example, up to 48.21% in the case of Spanish) if particular language characteristics are taken into account. Also in multilingual research, the FASTY system achieves keystrokes saving of above 50% for Dutch, French, German and Swedish with up to 15 proposals [10]. Moreover, using FASTY the writing speed (measured in words/minute) was found to increase only slightly within 5 weeks of writing.

The systems designed and implemented by the authors of this paper are presented and evaluated in [28]. The results for the Spanish language are up to 52.22% for keystroke saving, and 97.26% of the words may be predicted (with one, five and ten proposals). The results for the Basque language (highly inflected) are, with a maximum of ten proposals, up to 52.52% for keystroke saving and 90.09% of the words may be predicted. Keystroke saving is similar in both languages, with different hit ratios mainly due to the fact that the average length of words in Basque is higher than in Spanish.

9 User's effort

When designing prediction systems, there are factors that are very much related to the user's characteristics and must be evaluated to obtain the best possible results [65]. Otherwise, if an non-optimal solution is adopted, the added complexity may cause degradation in the use of the system. For example, as previously mentioned, the number of offered proposals influences search time (see Sect. 7.2). Thus, a high number of proposals may cause the system to become unusable in spite of the great increase in the hit ratio.

Ergonomic factors and system usability features have to be considered, bearing in mind the possible differences among the potential users. The interface must be adaptable enough in order to achieve the optimal configuration for each user in a given user environment. In this way, cognitive and physical effort associated to the system should be minimised and the effects related to fatigue reduced. Consequently, a more efficient communication speed may be achieved and the system may be used for a longer period of time. However, the scientific community does not appear to have achieved a consolidated view of these factors and how to address them. Many design decisions are based on the personal experience of the designers or on common knowledge of the user's needs.

There is a need for rigorous usability studies to support design criteria. If a number of factors are badly considered, the complexity increases, and the physical and the cognitive efforts required on the part of users are also higher. This can also cause fatigue to appear sooner and communication speed to decrease. Therefore, it is necessary to take the users' preferences into account when designing the system, as this may improve efficiency. It is also important for the system to be adapted to the users' individual characteristics, as discussed in Sect. 10. For example, Koester and Levine [47] suggest that reading the proposals in sequence (to find the appropriate one) requires less time when they are alphabetically ordered instead of being ordered by frequencies. Colouring the options by syntactic category (e.g. blue for verbs, green for nouns) may also enhance the search time for several users. Newell et al. claim that decisions taken while designing factors such as scanning speed, size and colour of the options, and the possibility of providing help and personalizing the characteristics of the system may increase user acceptance [56].

The use of a virtual keyboard (near to the list of proposals) on a touch screen may decrease the physical effort required, as there is no need for the users to move their head from the keyboard to the screen to find the required action [31].

Cognitive effort must be taken into account, as many users do not read the proposals if the cost of reading, searching and processing the options is significant. Similarly, a number of users do not use systems giving the option to accept partially correct proposals which have to be modified by the user [11, 29], due to the cognitive effort needed.

There is a need to make a number of intelligent decisions in order to adequately adjust predictor features [56] and decrease the cognitive and physical effort on the part of the users. In a related study, Wester discusses methods to test the functionality and usability of a word prediction prototype [66].

10 Adaptability in prediction

As previously mentioned, the possibility of making the predictive system adaptable to the user's characteristics and/or preferences may enhance the final results. In this section, a number of features that can be adapted are discussed; others have been presented in previous sections.

10.1 Adaptation of the number of proposals

An aspect that can be adapted in a prediction system is the number of proposals offered by the system [31].

The objective of making the maximum number of proposals adaptable is to minimise the required effort from users, and therefore the effects of fatigue, maintaining the message composition rate. To perform the adaptation, the system automatically decreases the number of proposals until the hit ratio achieved diminishes. In this way, the user will have to read less proposals and he/she will select the option he/she is going to write next more quickly (this will be even faster in a scan-based input, where less options are scanned faster).

However, it is difficult to obtain a suitable compromise among the number of proposals, the message composition rate and fatigue or error rate. Decreasing the number of proposals until the error rate increases also causes the decrease of the composition rate and of the hit ratio. Therefore, to the authors' knowledge, this type of adaptation is not widely used, due to the complexity of the interrelation among the involved factors and the fact that these systems normally use a fixed maximum number of proposals.

10.2 Adaptation of the scanning speed

If the scanning speed is too fast for a given user, the error rate will be high. That is, the probability that the required option is not properly selected is increased while increasing the speed. This situation can become quite stressing and cause user fatigue [32]. Furthermore, the frustration inherent to the high error rate when making selections may cause a user rejection on this type of systems. On the contrary, users can also reject very slow systems because of the required time needed to compose a message. This last factor can also cause an increase of fatigue.

Thus, there is a need to properly configure scanning speed. Most systems create a user profile and provide fixed values to the scanning rate (selected from a set of values). It is also possible to design adaptive systems that dynamically calculate the best value for each user by using user modelling techniques based on fuzzy logic [32].

10.3 Adaptation of the distribution of the options

When the predicted blocks are not words, they occupy fixed positions on the interface. This can help the user memorise these positions and therefore react more quickly and with a lower cognitive effort, that is, with less fatigue [47, 50].

When the predicted blocks are words, their probabilities can be different for diverse sentences, as adaptive dictionaries update the frequency of use of the words. In addition, if the system uses morpho-syntactic information, a given word will be proposed with a different probability, depending on its related morpho-syntactic information. Therefore, the guessed words can appear each time in a different order. Only in pure statistical systems, where no adaptation is made, the proposals may appear in fixed places and users memorise their locations.

Therefore, the distribution of the proposals is not usually fixed, requiring more attention and cognitive effort from users in order to find if the desired word is in the list of guessed blocks.

11 Special features

Prediction systems are reported in the literature which exhibit particular features not present in other predictors. This section reviews the most relevant of these features.

For example, the dictionary of the PAL system developed at the University of Dundee [60] has a tree structure. This structure proves to be extremely useful to gain access to the lexicon; however, the storage size and the management of the dictionary increase. Another factor also studied by this research team is the possibility of making remote prediction by using the client-server model and displaying the interface on the local machine [9]. The minimum bandwidth required for this objective is estimated to be 9,600 bauds. With a lower bandwidth, the prediction list requires a long time to arrive and the user tends to ignore it.

The script technique to make predictions has also been studied [56]. Scripts capture the essence of stereotyped situations and allow taking into account what it is happening and predicting what is going to happen next. The Script Talker system uses a scene-based interface to guide users through common dialogues.

There is a remarkable effort devoted to product speech acts in a competent way, instead of enhancing the production of letters, words or specific sentences [4, 5, 6]. A speech act is a phrase or a sentence that has a particular purpose in dialogue. The CHAT prototype is based on the patterns found in non-restricted dialogue. The main purposes of this technique are to increase the communication speed, to express the user's personality and way of communication, and to minimise silence.

The COHORT programme, described in [41], is especially designed for people with aphasia. This system tries to reconstruct the current word from phonemes, phonetic components or stressed syllables uttered by the user in an incorrect order. The proposals may be presented on the computer screen or by means of synthesised voice.

The PROFET system applied to Swedish, a highly inflected language, can make the prediction of a word in two steps: one for the word-root and the other for the suffix [11]. PROFET II includes syntactic and semantic information and uses Markov models to consider the previous last two words and their frequencies [14, 15, 53].

The FASTY system presented in [10] has several innovative features, such as prediction of compounds, prediction of proper inflectional forms based on the use of parsing, generic algorithms to ensure cross-language portability, dictionaries based on general language corpora and on users' own texts, adaptation of dictionaries based on the use of the predictor, several languages supported (initially, Dutch, French, German and Swedish), a user interface which is an integral part of the prediction system and new input devices.

There are several works focused on *n*-gram prediction. For example, the Reactive Keyboard system is devoted to predict text written in any natural language, programming language, command interpreter, etc. [19, 35]. Predictions are generated from an adaptive model obtained from previously written text. It uses a ''click and drag" interface, which makes extensive use of the mouse.

Lesher et al. study *k*-gram, *n*-gram and word prediction strategies in order to include them in scan-based interfaces [50]. Square matrixes are used to lay out the selection set. The predictor's proposals are located in these matrixes. Human word prediction is quantified in order to estimate the desirable performance [51].

Several systems make syntactic predictions using an X' grammar-based probabilistic ATN parser [63]. They analyse sentences top-down, expanding the grammar rules by level instead of by depth.

Claypool et al. present a statistic predictor based on word tri-grams (series of three words) applied to Gaelic, within a project relative to various European languages such as English, French, German, Spanish, etc. [16]. This predictor is called PredictAbility and its results are evaluated by programmes and stored in files compatible with MATLAB.

Wood and Lewis [67] use a grammar description language (GDL) to describe an aspect of English grammar and apply it to word prediction. Based on this idea, they have also developed a prototype called WindMill [68].

The VITIPI presented in [13] is a word predictor that, when no proposal may be presented, tries to infer certain analogies or even alter strings in order to present one.

The authors of [48] and [49] distinguish between manual (written) and oral (by speech recognition) word prediction and study where to apply each one.

In [58], artificial neural networks are used to make predictions for Spanish. A network is implemented using 130 neurones (100 in the input layer, 10 in the hidden layer and 20 in the output layer), which obtains the probabilities of the 20 categories of Part Of Speech (POS) categorisation.

Even if most systems are designed for the Windows environment [36], there are several for other platforms. For instance, [22] presents a method for the Linux operating system and [69] refers to a platform-independent system built in Java.

The authors of [24] present a chart bottom-up parser, which may be used in syntactic word prediction for Spanish. The results obtained with this technique (comparing it with the prediction using frequencies) suggested its application to a highly inflected language, such as Basque [29], in a two-step process to complete the words.

To conclude, several prediction systems allow for abbreviation expansion, topic-specific lists, word modifications and speech features [54].

12 Conclusions

Within AAC systems, text prediction techniques have been frequently designed with the aim of increasing the communication rate. These techniques have been initially oriented to people with severe motor and oral disabilities, but non-disabled people can also use them when composing messages as they can be seen as intelligent agents. Several authors proposed very different predictive techniques and strategies that are difficult to compare.

This paper has presented a study of text prediction and its related factors. The factors analysed are text block size, dictionary features, prediction method, influence of the language and the user interface, usability, adaptability, and other special features.

The various prediction measurements and their results have been shown, both objectively (using programmes emulating the user's behaviour) and subjectively (by means of user tests). The results are mainly expressed through keystroke saving. However, other factors such as the hit ratio may be considered. The most relevant results found in the literature have been referred.

In order to compare the different techniques in an appropriate way, the need for a standardised workbench is pointed out. Without standard tests the comparisons made are merely informative and not provable, and they cannot be used to establish whether one approach is better than another.

Additionally, this paper has addressed the user interface issues in prediction systems, as they have a huge influence on the predictor's performance, as well as on the usability of the system. Related important factors, such as the number of proposals and their location, the acceptation/rejection protocol, etc., have been discussed, taking into account the experiences reported in the literature. Nevertheless, there is no unique configuration of the parameters that satisfies the requirements of the entire user population, mainly due to the physical and cognitive diversity of the users. For this reason, the paper has provided general guidelines that have to be considered by designers.

In addition to the mentioned design guidelines, it is necessary to perform usability and ergonomic studies in order to select the most suitable system features for a given user. Even a very efficient predictive system may be useless for a specific type of user if it does not consider their special characteristics.

References

1. AAC (1989) Augmentative and alternative communication 5(1)
2. Abascal J, Gardeazabal L, Garay N (2004) Optimisation of the selection set features for scanning text input. ICCHP 2004, LNCS 3118. Springer, Berlin Heidelberg New York, pp 788–795
3. Agirre E, Alegria I, Arregi X, Artola X, Diaz De Ilarraza A, Maritxalar M, Sarasola K, Urkia M (1992) Xuxen: a spelling checker/corrector for Basque based on two-level morphology. In: Proceedings of the 3rd conference on applied natural language processing. Association for Computational Linguistics, pp 119–125
4. Alm N, Arnott JL, Newell A (1989) Discourse analysis and pragmatics in the design of conversation prosthesis. J Med Eng Technol 13(1/2):10–12

5. Alm N, Newell A, Arnott J (1989) Revolutionary communication system to aid non-speakers. Speech Therapy in Practice Update, pp VII–VIII
6. Alm N, Arnott JL, Newell AF (1992) Prediction and conversational momentum in an augmentative communication system. Comm ACM 35(5):46–57
7. Arnott JL, Javed MY (1992) Probabilistic character disambiguation for reduced keyboards using small text simples. Augmentative Alternative Commun 8:215–223
8. Arruabarrena A, González J (1988) Comunicadores y Emuladores de Teclado. Tecnologías de la Información y Discapacidad. Fundesco, Madrid, pp 133–156
9. Beattie W, Hine N, Arnott J (1998) Distributed assistive communication devices for non-speaking users. In: Proceedings of the Isaac conference, Dublin (Ireland), pp 447–448
10. Beck C, Seisenbacher G, Edelmayer G, Zagler W (2004) First user test results with the predictive typing system FASTY. ICCHP 2004, LNCS 3118. Springer, Berlin Heidelberg New York, pp 813–819
11. Bertenstam J, Hunnicutt S (1995) Adding morphology to a word predictor. In: The European context for assistive technology (Proceedings of the 2nd tide congress). IOS Press/Ohmsa, Bruselas, pp 312–315
12. Beukelman DR, Yorkston KM, Poblete M, Naranjo C (1984) Frequency of word occurrence in communication samples produced by adult communication aid. J Speech Hear Disord 49:360–367
13. Boissiere P, Dours D (1996) VITIPI: versatile interpretation of text input by persons with impairments. In: Interdisciplinary aspects on computers helping people with special needs: 5th international conference (ICCHP'96). Oldenbourg, Linz (Austria), pp 165–172
14. Carlberger J (1997) Design and implementation of a probabilistic word prediction program. Master's Thesis in Computer Science, Nada, KTH, Stockholm (Sweden). http://www.speech.kth.se/~Johanc/Thesis/Thesis.doc
15. Carlberger A, Carlberger J, Magnuson T, Hunnicutt MS, Palazuelos-Cagigas SE, Aguilera Navarro S (1997) Profet, a new generation of word prediction: an evaluation study. In: Proceedings of the natural language processing for communication aids workshop. Association for Computational Linguistics, pp 23–28
16. Claypool T, Ricketts I, Gregor P, Booth L, Palazuelos S (1998) Learning rates of a tri-gram based gaelic word predictor. In: Proceedings of ISAAC conference, Dublin (Ireland), pp 178–179
17. Copestake A (1997) Augmented and alternative NLP techniques for augmentative and alternative communication. In: Proceedings of the natural language processing for communication aids workshop. Association for Computational Linguistics, pp 37–42
18. Copestake A, Flickinger D (1998) Evaluation of NLP technology for AAC using logged data. In: Isaac 98 research symposium proceedings, Whurr Publishers, London. http://www-csli.stanford.edu/~aac/evaluation-dsp1.htm
19. Darragh JJ, Witten IH, James ML (1990) The reactive keyboard: a predictive typing aid. Computer 23(11):41–49
20. Digiovanni JM (1996) Word prediction software as a tool to improve writing literacy. http://san183.sang.wmich.edu/Sped603fall96/Digiovannipaper.html
21. Even-Zohar Y, Roth D (2000) A Classification approach to word prediction. In: Proceedings of the Naacl'00. http://l2r.cs.uiuc.edu/~Evenzoha/Papers/Classificationapproach.Ps
22. Gaiarin M (1999) Un Sistema di Predizione Sintattica per Gnu/Linux. Tesi Di Laurea. Università Degli Studi Di Padova. http://www.dei.unipd.it/~Gaio/Tesi/Favele.Ps.Gz
23. Garay N, Abascal J (1994) Using statistical and syntactic information in word prediction for input speed enhancement. Information Systems Design and Hypermedia. Cépaduès-Éditions, Toulouse (France), pp 223–230
24. Garay-Vitoria N, Abascal J (1994) Application of artificial intelligence methods in a word-prediction aid. ICCHP 1994, LNCS 860. Springer, Berlin Heidelberg New York, pp 363–370
25. Garay-Vitoria N, Abascal J, Urigoitia-Bengoa S (1995) Application of the human conversation modelling in a telephonic aid. In: Proceedings of the 15th international symposium on human factors in telecommunications (HFT'95), Melbourne (Australia), pp 131–138
26. Garay-Vitoria N, Abascal J (1997) Intelligent word-prediction to enhance text input rate. In: Proceedings of the intelligent user interfaces 97 congress. ACM Press, New York, pp 241–244
27. Garay-Vitoria N, Abascal J (1997) Word prediction for inflected languages. Application to Basque language. In: Proceedings of the natural language processing for communication aids workshop. Association for Computational Linguistics, pp 29–36
28. Garay-Vitoria N (2001) Sistemas de Predicción Lingüística. Aplicación a Idiomas con Alto y Bajo Grado de Flexión, en el Ámbito de la Comunicación Alternativa y Aumentativa. Servicio Editorial de la Universidad del País Vasco, Leioa (Spain). ISBN:84–8373–355–2
29. Garay N, Abascal J, Gardeazabal L (2002) Evaluation of prediction methods applied to an inflected language. In: Proceedings of the 5th International Conference Text, Speech and Dialogue TSD2002. Springer, Berlin Heidelberg New York, pp 397–403
30. Garay N, Abascal J (2004) A comparison of prediction techniques to enhance the communication rate. In: Proceedings of the 8th ERCIM workshop, LNCS 3196. Springer, Berlin Heidelberg New York, pp 400–417
31. Garay-Vitoria N, Abascal J (2004) User interface factors related to word prediction systems. In: Proceedings of the 7th international conference on work with computing Systems WWCS2004. Damai Sciences, Kuala Lumpur (Malaysia), pp 77–82
32. Gardeazabal L (2000) Aplicaciones de la Tecnología de Computadores a la Mejora de la Velocidad de Comunicación en Sistemas de Comunicación Aumentativa y Alternativa. Servicio Editorial de la Universidad del País Vasco, Leioa (Spain). ISBN:84–8373–301–3
33. Gillette Y, Hoffman JL (1995) Getting to word prediction: developmental literacy and AAC. In: Proceedings of the 10th annual international conference, technology and persons with disabilities. Northridge, California. Taken From [54]
34. González Abascal J (1992) Un Método de Compresión de Textos Basado en la Posicionalidad y el Rendimiento. Aplicación al Euskara. Servicio Editorial de la Universidad del País Vasco, Leioa (Spain). ISBN:84–7585–301–3.
35. Greenberg S, Darragh JJ, Maulsby D, Witten IH (1995) Predictive interfaces: what will they think of next? Extra-ordinary human-computer interaction. Interfaces for users with disabilities, Edwards ADN (Ed) Cambridge University Press, London, pp 103–140
36. Hecht J, Heinisch B (1997) A comparative overview of word prediction software for Windows 95. http://www.southernct.edu/Departments/Cat/Wordpred.html
37. Heckathorne CW, Childress DS (1983) Applying anticipatory text selection in a writing aid for people with severe motor impairment. IEEE Micro 3(3):17–23
38. Heinisch B, Hecht J (1993) Predictive word processors: a comparison of six programs. Tam News 8:4–9
39. Higginbotham D (1992) Evaluation of keystroke savings across five assistive communication technologies. Augmentative Alternative Commun 8:258–272
40. Hortsmann HH, Levine SP (1990) Modelling of user performance with computer access and augmentative communication systems for handicapped people. Augmentative Alternative Commun 6:231–241
41. Hunnicutt S (1987) Input and output alternatives in word prediction. STL/QPRS 2–3/1987, pp 15–29

42. Hunnicutt S (1989) Using syntactic and semantic information in a word prediction aid. In: Proceedings of the European conference on speech communication and Technology 1, Paris, pp 191–193
43. Hunnicutt S, Nozadze N, Chikoidze G (2003) Russian word prediction with morphological support. In: 5th International symposium on language, logic and computation, Tbilisi, Georgia
44. Klund J, Novak M (2001) If word prediction can help, which program do you choose? Retrieved February 16, 2004, from http://www.tracecenter.org/docs/wordprediction2001/
45. Koester HH, Levine SP (1994) Modelling the speed of text entry with a word prediction interface. IEEE Trans Rehabil Eng 2(3):177–187
46. Koester HH, Levine SP (1996) Effect of a word prediction feature on user performance. Augmentative Alternative Commun 12:155–168
47. Koester HH, Levine SP (1998) Model simulations of user performance with word prediction. Augmentative Alternative Commun 14:25–35
48. Laine CJ (1998) Using word-prediction technology to improve the writing of low-functioning children and adults. In: Proceedings of the Australian computers in education conference. http://www.cegsa.sa.edu.au/Acec98/Papers/P-Laine.html
49. Laine CJ, Bristow T (1999) Using manual word-prediction technology to cue student's writing: does it really help? CSUN99. http://www.dinf.org/Csun_99/Session0067.html
50. Lesher GW, Moulton BJ, Higginbotham DJ (1998) Techniques for augmenting scanning communication. Augmentative Alternative Commun 14:81–101
51. Lesher GW, Moulton BJ, Higginbotham DJ, Alsofrom B (2002) Limits of human word prediction performance. 2002 technology and persons with disabilities conference, California State University, Northridge
52. Magnuson S (1995) Word prediction as linguistic support for individuals with reading and writing difficulties. In: Placencia Porrero I, Puig de la Bellacasa R (eds) The European context for assistive technology. IOS Press, Ohmsa, pp 320–323
53. Magnuson T (1998) Linguistic Evaluation of PROPHET II: a pilot project. In: Proceedings of the Isaac conference, Dublin (Ireland), pp 479–480
54. National Center to Improve Practice (2004) NCIP Library. http://www2.edc.org/Ncip/Library/Toc.htm
55. Newell AF, Arnott J, Waller A (1992) On the validity of user modelling. In: AAC: comments on Horstmann and Levine (1990) Augmentative Alternative Commun 8:89–92
56. Newell AF, Arnott JL, Booth L, Beattie W, Brophy B, Ricketts IW (1992) Effect of the "Pal" word prediction system on the quality and quantity of text generation. Augmentative Alternative Commun 8:304–311
57. Palazuelos-Cagigas S, Aguilera S, Claypool T, Ricketts I, Gregor P (1998) Comparison of two word prediction systems using five European languages. In: Proceedings of the ISAAC conference, Dublin (Ireland), pp 192–193
58. Palazuelos-Cagigas S, Aguilera S, Ricketts IW, Gregor P, Claypool T (1998) Artificial neural networks applied to improving linguistic word prediction. In: Proceedings of the ISAAC conference, Dublin (Ireland), pp 194–195
59. Shannon CE (1951) Prediction and entropy of printed English. Bell Syst Techn J 30(1):50–64
60. Swiffin AL, Arnott JL, Pickering JA, Newell AF (1987) Adaptive and predictive techniques in a communication prosthesis. Augmentative Alternative Commun 3:181–191
61. Swiffin AL, Arnott JL, Newell AF (1987) The use of syntax in a predictive communication aid for the physically handicapped. In: RESNA 10th annual conference, San Jose (California), pp 124–126
62. Tegic Communications (2004) T9 Text Input for Keypad Devices. http://www.tegic.com
63. Van Dyke JA (1991) Word prediction for disabled users: applying natural language processing to enhance communication. Thesis Submitted For The Degree Of Honors Bachelor of Arts In Cognitive Studies. University of Delaware
64. Venkatagiri HS (1993) Efficiency of lexical prediction as a communication acceleration technique. Augmentative Alternative Commun 9:161–167
65. Venkatagiri HS (1999) Efficient keyboard layouts for sequential access in augmentative and alternative communication. Augmentative Alternative Commun 15:126–134
66. Wester M (2003) User evaluation of a word prediction system. Master's thesis in computational linguistics, Uppsala University, Uppsala (Finland). http://stp.ling.uu.se/ ~matsd/thesis/arch/2003_wester.pdf
67. Wood MEJ, Lewis E (1993) Grammatical recognition in computer aided conversation. In: Proceedings of the European speech communication association workshop on speech and language technology for disabled people. Stockholm (Sweden), pp 115–118
68. Wood MEJ (1996) Syntactic pre-processing in single-word prediction for disabled people. PhD Thesis of the Department Of Computer Science. University of Bristol
69. Worthylake K (2000) A platform-independent word prediction editor. Honours Thesis Defence. Acadia University, March. http://cs.acadiau.ca/Seminars/Previous/Kevinworthylake.html
70. Zordell J (1990) The use of word prediction and spelling correction software with mildly handicapped students Taken From [54]. Closing the Gap 9(1):10–11