Pertti A. Väyrynen, Kai H. Noponen and Tapio K. Seppänen

Multi-word Prediction for Legal English Context: A Study of Abbreviated Codes for Legal English Text Production

Abstract

Here, we investigate using strings of expressions longer than a single orthographic word in English word prediction in the legal English domain. The goal of the kind of prediction strategy, called multi-word prediction, is to speed up performance of humans in text production by means of word prediction. Accuracy of two prediction techniques was preliminarily estimated on a simulation without using human subjects with the lexicon of 7,009 multi-word units of legal English. The results show that the average of 70% of characters can be saved for the units in the lexicon in the best-case performance. An improvement in performance actually gained with a real text mainly depends on length and token frequency of units predicted. We also show how the length of multi-word units predicted appear to be related to the code lengths used in their prediction and how this finding can be utilized to practical advantage in multi-word prediction.

1. Introduction[1]

Mainstream typing assistant systems based on *word prediction* (e.g. WordQ) are mainly word-based. Such application programs aim at speeding up the performance of humans in text production by attempting to save both the number of characters/keystrokes (effort savings), or time required to type the text in (time savings). Usually, these applications process only one word token at a time by means of single word prediction. Along with single word tokens, however, texts also contain longer, more or less fixed, multi-word units such as collocations, semi-fixed phrases, idioms, lexical bundles (Biber et al. 1999), etc., which could also be looked up as single units in word prediction, resulting in greater effort and time savings, at least in theory.

[1] This article contains text that has previously been published in a doctoral dissertation available in Acta Universitatis Ouluensis Series. A full reference is given in references.

In the present study, we preliminarily investigate the potential of *multi-word prediction* for achieving an improvement in performance in English word prediction in a specific domain, legal English. The term multi-word prediction means here finding sequences of words consisting of two or more tokens using word-initial characters as codes (search keys, abbreviations) for their prediction. Although word prediction systems are not primarily made use of in this domain, legal English contains many formulaic multi-word units suitable for an initial estimation of the maximum utility of multi-word prediction under optimal conditions, that is, the best-case performance for the type of prediction methods proposed here.

An approach such as this to word prediction is motivated by the fact that, according to Jackendoff (1997: 156), a major part of language, in fact, consists of multi-word units of some sort: in English, for example, the number of such units equals that of single words in a person's lexicon. This is additionally confirmed by the contents of on-line lexical resources such as WordNet 1.7, for instance, where 41% of the entries are multi-word units of some sort (Villavicencio et al. 2005). To our knowledge, the performance of prediction methods tested here may not yet be reported in the literature, although the CHAT system presented in Alm et al. (1992) can be regarded as a precursor of a multi-word predictor. Some aspects of multi-word prediction, called *phrase prediction*, were also investigated in Väyrynen (2005) and Väyrynen et al. (2007).

More specifically, we are investigating a prediction strategy in which a user can retrieve or access strings of expressions longer than a single orthographic (phonological) word from a pre-registered set of such units by using word-initial characters as codes for their prediction. The prediction mechanism proper for finding units of this kind is based on *abbreviation expansion*, where any combination of initial characters of some tokens of the multi-word unit (not necessarily adjacent to each other) can be used as a code to access the units predicted.

As language users we are familiar with such abbreviations as ASR for expressions such as *automatic speech recognition*. It is the same kind of abbreviations the use of which is proposed here for multi-word prediction. Whether people are actually using codes of this sort in lexical access or retrieval of multi-word units, or more importantly, could learn to use them in accessing units of this kind in word prediction, is a psycholinguistic, empirical question that remains to be tested in further research.

To illustrate, to look up a two-word unit such as *legal tender*, for instance, the following codes consisting of word-initial characters could be used in prediction: *l,le*; *lt, let, lete,* etc. Along with the number of such units in English, an approach of this kind to multi-word prediction is additionally motivated by the fact that abbreviation expansion is known to be one feature of text inputting methods (one of which is word prediction) that is appreciated most by professional writers. As a consequence, it can be an asset in a practical prediction system assisting text production, especially if the prediction technique proposed turns out to be realistic enough for practical use. The net gain to a user of a prediction system of this type is the savings of efforts in terms of characters/keystrokes saved and the resulting time savings when using multi-word prediction, rather than writing the same units without the help of word prediction.

As usual in word prediction, some time will always be spent on finding the correct prediction on the word list (prediction list), where units matching a given code will be suggested to the user. The list of five units is used most often in practical prediction systems to minimise the time spent on finding the correct prediction on the list, still yielding relatively good character savings.

When using single initial codes such as *pp* for *parliament president*, for instance, the number of *homographic* codes matching more than one unit depends on the number of such units in the lexicon. In theory, the maximum number of two-word units predictable with 26 character alphabet in English using only one word-initial character as a code is 676 (= 26^2). As seen in Table 1, there are 5,698 two-word units in our lexicon of some 7,000 multi-word units of legal English, only 179 of which can be predicted using only one initial character of the two adjacent tokens as a code, e.g. *lt* for *legal tender*. In practice, a longer code containing two or more initial characters will have to be used in prediction.

In the empirical part, the performance of two prediction methods was investigated: one in which a user can type the first character of each word in a multi-word unit (e.g. *pp* for *parliament president*) and another, where the user types two letters per word (e.g. *papr* for *parliament president*). In longer multi-word units, however, as in *legal conceptual definition*, for example, the first (few) word-initial characters of any token of the unit can be used as a code; thus, *leco* or *lede*, for instance, could be employed in the above example. It should be noted that this is a more flexible way to use abbreviations than the similar procedure of abbreviation expansion available in word processors, where the abbreviations used must be predetermined and remembered for their use in practice.

As usual in word prediction experiments, evaluations of the two prediction scenarios investigated here were conducted on a simulation with a pre-defined set of multi-word units in the legal English domain without using human subjects. The results of simulation experiments thus imitate the typing performance of a perfect user who knows the shortest abbreviation or code for accessing a given multi-word unit. This was done to find out the maximum theoretical utility of the prediction techniques proposed here first. After that, we can estimate its practical utility with real texts and users, left for further research.

It should be noted that in order to use a practical prediction system with a multi-word prediction utility, the user is *not* expected to know beforehand the best possible code for finding a given unit, nor the exact contents of the lexicon of such units, for that matter. A system of this kind can still be useful to its users, regardless of the fact that the practical accuracy of multi-word prediction will naturally be somewhat lower than its theoretical performance due to the usage of less than optimal codes.

The structure of the remaining part of the article is as follows: first, word prediction will briefly be introduced in general terms in the following section. Both traditional prediction methods and multi-word prediction will then be dealt with in more detail, including our assumptions about multi-word prediction as an alternative prediction technique. Section 3 is concerned with evaluation protocols in word prediction and their limitations, followed by the empirical evaluation of two multi-word prediction methods in the following section.

The main issue of our preliminary results of the prediction experiments with multi-word units reported here is how to generalize from pre-stored abbreviations for stock phrases to a much more flexible and ad hoc kind of abbreviation expansion for multi-word units. We also test how the length of multi-word units predicted appears to be related to the code lengths (Figure 1) and how this finding could be utilized to practical advantage in multi-word prediction.

2. Word prediction

Word prediction can be used to aid text production by people experiencing various sorts of disabilities or physical or sensory restrictions. It can, however, be also useful for any writer, especially in devices that do not feature full-sized keyboards.

It is, however, likely that the multi-word prediction strategy proposed here is more suitable for users without disabilities or breakdowns in writing ability. As a new application area for multi-word prediction, Langlais et al. (2002) suggest using it as an embedded utility in a machine translation system. The greatest technological challenge in word prediction is caused by the fact that it is only the left context that is available for the prediction of a word token (or a longer unit) in a given context of use, which unfortunately, is often insufficient for prediction purposes.

Current prediction systems exploit a *language model*, which attempts to capture regularities in natural language in order to improve the performance of a variety of practical language technology applications, including word prediction. As in many fields of use in present language technology, machine translation, document classification, and information retrieval, to mention only a few (Rosenfeld 2000: 1), it is the so-called *n*-gram language models that are typically made use of also in word prediction. The *n*-gram stands for a sequence of *n* consecutive items, which can be letters, parts of speech, or words.

A key issue in language modelling is *smoothing* (see e.g. Chen & Goodman 1996), handling the problem of sparse data (characteristic of natural language) when creating language models. By means of smoothing, statistical *n*-gram language models can be made more robust to alleviate the problem of non-occurrence of all possible word *n*-grams (or multi-word units) in the training corpus no matter how large it is. Since some of our most widely used techniques of language modelling, simple *n*-gram language models for one, have already reached their maximum limit of performance (Rosenfeld 2000), some new ways will have to be found for improving the accuracy of our future word prediction systems further. One alternative prediction method worth investigation is thus the prediction of multi-word units.

In what follows, a typology of word prediction methods will be presented; a more detailed survey of them can be found in Garay-Vitoria and Abascal (2006): word tokens can traditionally be predicted in two ways: in *word completion*, tokens are predicted by typing in one or more initial characters to the code.

Possible word tokens appropriate for a given context of use can also be predicted on the basis of the linguistic context of *n*-1 preceding tokens using *n*-gram language models (Cook & Hussey 1995). The two prediction methods described above are the basic, well-established prediction techniques used (and investigated most thoroughly) in practically all modern (statistical) word prediction systems. Multi-word prediction, in its turn, is a newer prediction method, less thoroughly investigated, especially as outlined here.

To improve the performance of prediction systems further, other additional prediction techniques can also be made use of in the same prediction system. These include the so-called *recency of usage* (*recency of mention, recency promotion*) (Carlberger 1998; Swiffin et al. 1987), in which a larger history of preceding words, say, 40 tokens is considered in prediction. This prediction method, modeling the tendency of previously used words to recur within a given word history in the text due to anaphora, for example, can be quite effective in practice. For example, in an English corpus Rosenfeld (1996) analyzed, the best predictor of identical lexical repetition turned out to be the word itself in 65% of the cases; in 90% of the cases, the word itself was among the six best predictors. According to him, word tokens having the same stem are also good predictors of each other.

Besides recency promotion, another prediction technique employed in commercial word prediction at present is (intelligent) *abbreviation expansion* (McCoy & Demasco 1995), where a user defines abbreviations in the system's set of abbreviations in advance. Every time the user types an abbreviation such as *goo* for *good morning*, the system simply replaces it by the original text. The example above represents one type of multi-word prediction which uses *n*-gram language models, predicting the rest of the unit from its onset. Abbreviations such as ASR for *automatic speech recognition* can also be used in the traditional abbreviation expansion. It would be interesting to know how the performance of this kind of multi-word prediction and the one proposed here will differ in practice. As mentioned above, the drawback of this method is, however, that the abbreviations will have to be predefined in the prediction system and remembered by the user for their usage.

It should be noted that this is not the case regarding the prediction strategy proposed here, where any combination of word-initial characters of some tokens of a multi-word unit can be used as a code more flexibly.

As for the performance of word prediction systems in general, known as their *prediction accuracy*, results reported in the literature appear to vary, dependent on text type, prediction method (or a combination of them), and the test corpus used. To illustrate, according to Matiasek et al.

(2002: 1), the percentage of *keystroke savings* of state-of-the-art prediction systems can be as high as 75% with more than one prediction method in the same system. For WordQ predictor, the keystroke savings rates varied from 37% to 53% with three test texts, containing 116,578 word tokens in all (Nantais et al. 2001). In Wood's Windmill system (1996), the same rates varied between 30.4% and 55.1%, depending on text type and prediction algorithm. As reported in the literature, about 50% of characters/keystrokes can be saved on average in statistical state-of-the-art prediction systems, which, of course, does not equal the time savings achievable in text production by means of word prediction, discussed in more detail below.

It should be noted that single word prediction can serve the purpose of multi-word prediction as well. As a result, any multi-word unit, *legal tender*, for one, can always be predicted by means of single word prediction, word by word, i.e. by predicting the word *legal* first and after that *tender*. Given the frequency of occurrence of multi-word units in language in general, it could be assumed that predicting such units as one unit will improve the accuracy of word prediction systems further. What is more, similar units could naturally be predicted in any domain in many languages, provided that they are available in the lexicon.

3. Evaluation protocols in word prediction and their limitations

In order to evaluate a word prediction system or new prediction technique, both *qualitative* and *quantitative* evaluation is required in practice with real users representing the target user group. The former can cover a detailed analysis of the text produced by means of word prediction, for example.

Accuracy of a prediction system is usually evaluated quantitatively by means of global measures of performance such as savings in keystrokes or characters (Wester 2003: 16). More often than not, however, as in this study, the performance of a perfect user is simulated in practice to estimate the accuracy of a prediction system objectively.

Regardless of their shortcomings such as hiding rather than displaying the details of the functioning of the prediction system (Väyrynen et al. 2007), simulations of word prediction performance with respect to effort savings obtainable are widely used in word prediction experiments and are practically useful for system development. For many, the real purpose of word prediction boils down to time savings achievable in text production by means of word prediction, however. Unfortunately, time savings are difficult to determine in practice due to many user characteristics, and are usually estimated as a factor proportional to keystroke savings. (Garay-Vitoria & Abascal 2006: 197.)

As mentioned above, the results of various prediction experiments available in the literature are not directly comparable to each other. As a result, only a rough comparison can be made in practice because of a diversity of prediction methods available and lack of a standard workbench. (Garay-Vitoria & Abascal 2006: 196–197.) Moreover, factors not directly related to the quality of word prediction may also affect the results achieved, including differences among languages or different performance measurements employed, for example (Palazuelos Cagigas 2001). Also, what should actually be measured generally depends on the prediction scenario envisaged.

Regarding the strategy of multi-word prediction proposed here, we investigate the following aspects of its performance:

(1) effect of the length of multi-word unit predicted on the number of units that can be predicted, using either one or two word-initial characters of some tokens of the units as a code (section 4.3);

(2) accuracy of multi-word prediction under optimal conditions for a perfect user given as the average percentage of characters saved for the multi-word units included in the lexicon of 7,009 units of legal English (section 4.3);

(3) effect of the physical cost of one keystroke on the percentage of characters saved on average when manually selecting the prediction mode for multi-word prediction in the interface of a hybrid prediction system, with a multi-word prediction utility along with single word prediction (section 4.3);

(4) how the token frequency of multi-word units in text appears to affect savings in characters achievable by means of multi-word prediction. This part of the study is in part based on the findings of Erman and Warren (2000), who attempted to quantify the proportion of a sample of text that is accounted for by multi-word-like entries (section 4.4).

Another alternative would be to investigate multi-word prediction using *n*-gram language models. In that case, the user would type in a few characters from the onset of the first token of the unit, e.g. *go* for *Good Morning*, as a code for prediction. This kind of prediction strategy was investigated by Eng and Eisner (2004) in another special field, the radiology report domain. An approach such as this to multi-word prediction was not chosen, however, because, for one thing, only one or more initial characters of the first token of a multi-word unit are typically employed as a code in this prediction method. For another, using a code of this type would also increase the number of matching units for units with the same premodifier. To illustrate, in our lexicon of multi-word units, the contents of which will be introduced in section 4.1, the adjective *legal*, for instance, appears in no less than 113 two-word multi-word units as a premodifier. As a result, in this case, there would be 113 possible units matching the code *leg*, for instance, in *n*-gram based prediction. We therefore opted for a more flexible prediction method, allowing the usage of any combination of initial characters for predicting multi-word units.

The main purpose of the research carried out here is to justify further research on an alternative prediction strategy. Therefore, the initial estimations of the average percentages of character savings given in section 4.3 are made very roughly, their shortcomings including the blindness to the visual-cognitive loads of using multi-word prediction due to finding the correct unit on the word list and heavy reliance on a perfect user (see Table 3, given as Appendix 1). In the future, we attempt to make more realistic estimations of the performance of multi-word prediction and study different aspects of it more thoroughly with real texts and users.

4. Empirical evaluation of two methods of multi-word prediction

Here we attempt to preliminarily estimate the potential of multi-word prediction for improving word prediction performance in English word prediction. Found useful enough, it could then be made use of as an alternative/additional prediction method alongside more traditional prediction methods for single word prediction in a hybrid prediction system, with separate prediction techniques for both single word prediction and multi-word prediction.

As regards multi-word prediction, an attempt is made to answer the following three research questions: (1) how should the prediction list be ordered for the best possible performance with respect to effort savings in multi-word prediction, (2) what factors appear to affect the performance of multi-word prediction, (3) what sort of character savings can be obtained under optimal conditions in multi-word prediction? The research questions (1) and (2) are interrelated, of course. As mentioned above, two techniques of multi-word prediction will be tested: first, using one initial character of some token of the multi-word unit, second, using two initial characters per word, respectively.

4.1 Lexicon

In general, a language can have a great potential for multi-word prediction, at least in theory. This, on the one hand, depends on the number and kind of multi-word units which actually appear in a predicted text, on the other, the coverage of the lexicon of such units employed in prediction, that is, whether or not it contains (ideally all) or the majority of the units predicted appearing in the text representing a given genre.

In the present study, a set of 7,009 multi-word units of legal English was collected for prediction experiments proper. In all, the data file contains 15,737 word tokens; the number of word types being 3,314. The items selected represent a small subset of single words and multi-word units included in the entire *English-Finnish Law Dictionary* by Joutsen (1985). The main selection criterion for the units included in the lexicon was their length with respect to the number of word tokens they contain, roughly corresponding to the length of such units in the text predicted, with most units being short ones (Biber et al. 1999: 597; Erman & Warren 2000). We also wanted to have a fairly large collection of such units. Table 1 shows the distribution of units in the lexicon by their length by means of the number of words they include.

Table 1. Distribution of multi-word units in the lexicon by length (number of words).

Number of word tokens in a multi-word unit	Number of multi-word units in the lexicon
2	5,698
3	1,015
4	217
5	55
6	18
7	3
8	3
Total	7,009

As seen in Table 1, most units consist of only two tokens (81.3%). The mean length of the unit is 2.25 tokens. The units chosen average 16.32 characters, including the white space between the words of the unit. The data file is arranged alphabetically. It should be noted that the results of the prediction experiments with multi-word units reported on below are highly genre-specific, representing legal language. Results in other application areas would certainly be very different.

4.2 Procedure

In all prediction experiments with multi-word units, it is assumed that a perfect user always knows the best possible code, i.e. the minimum number of word-initial characters for predicting a given unit. This is to determine the maximum prediction accuracy obtainable in multi-word prediction with our lexicon of multi-word units. Findings given in Figure 1 and Table 2 therefore result from trying all relevant codes and selecting the one that works best. For that purpose, we wrote a special algorithm for testing all possible codes in the prediction of individual units and selecting the best of them for their prediction. However unrealistic the above assumption may be in practice it, nevertheless, shows the best-case performance for the two multi-word prediction techniques tested here. After that, we evaluate what kind of prediction accuracy could be achieved in that kind of word prediction method with a real text with respect to the token frequency of multi-word units in the text predicted (see, section 4.4).

As mentioned, the performance of a word prediction system is usually quantitatively evaluated in terms of the number of keystrokes/characters saved in typing of text. The percentage of characters saved in multi-word prediction can be calculated as follows: let us define

c = the number of characters needed to predict a multi-word unit, including the internal blanks between the word tokens of a unit;
L = the length of the unit, including internal blanks;

$t = \begin{cases} 1 & \text{for multi-word prediction} \\ 0 & \text{for word completion} \end{cases}$ this is the one keystroke spent in switching from word completion in single word prediction to multi-word prediction.

Thus, we get $s = 1 - \frac{c+1+t}{L+1}$.

The quotient in the expression above is the percentage of characters required to type a multi-word unit. The one in the numerator represents either the one keystroke used in the selection of the multi-word unit or the white space typed by the user when the system cannot predict any such unit. The one in the denominator takes the automatically added blank into account. For multi-word prediction, *t* adds the one keystroke which is needed to toggle on multi-word prediction mode.

4.3 Savings in characters in multi-word prediction

Before estimating the character savings achievable on average under optimal conditions in multi-word prediction, we first investigate how the length of a multi-word unit may affect the number of units that can be predicted with a given prediction method. Figure 1 shows how multi-word unit length affects the number of (un)predictable units with the word list (prediction list) of one and ten units. Since using a word list of five units is unlikely to give any new information, this size was not used at all.

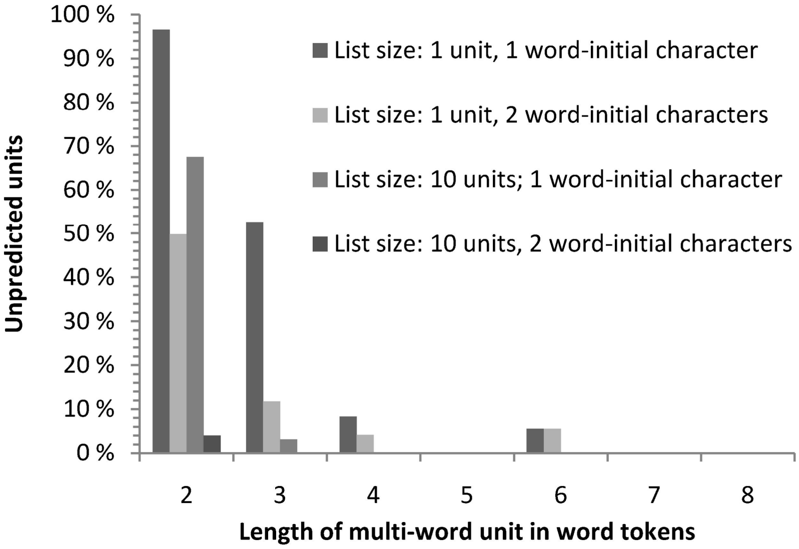

Figure 1. Effect of prediction list size and code type on the percent of multi-word units that cannot be predicted.

Figure 1 indicates the percentage of multi-word units that cannot be found with a given code. As seen, for shorter units of two word tokens only, for instance, 96.6% (5,502) cannot be predicted at all using only one word-initial character as the code with the prediction list of one token; 17 can be predicted using one initial character as the code; and 179 can be found using two initial characters as the code (the exact figures are derived from a larger table, a fraction of which is given here in Figure 1).

As seen in Figure 1, both prediction methods can be used to predict all multi-word units longer than three tokens with the prediction list of ten units, for which the number of units that cannot be predicted is zero. It should be noted, however, that the average length of the code is about one character shorter for the prediction method of one initial character with this list. This is also true of the prediction case with the list of only one item. However, this method cannot find all units. Unfortunately, there are rather few units longer than four word tokens in our lexicon of multi-word units (79 in all). The results above are not therefore very reliable statistically because of the lack of longer units in the lexicon. On the other hand, the distribution of the length of the units in our lexicon of multi-word units is probably typical, in the sense that the number of shorter units is larger than that of longer ones, also corroborated by the findings of Erman & Warren (2000: 40).

Based on the findings of Figure 1, “a mix of search techniques” appears to be the most efficient prediction method with respect to the length of the multi-word unit predicted: the user needs to type two word-initial characters of the first one or two tokens of the unit for shorter units consisting of two or three tokens. For example, the code *leco* could be used for *legal competence*. In that case, the code length is either two or four characters, respectively. For units longer than three tokens, the user can type in just one initial character of the first three adjacent tokens of the unit, for example, *lca* for *letter containing an order*. A further advantage of mixing the two prediction techniques is that it will also maximize the prediction accuracy achievable in multi-word prediction, that is, the average percentage of characters saved.

What is more, the length of the code could also be employed as a *cue* providing information on the type of multi-word unit predicted: the even number of characters in the code for predicting two-word units, the odd number for units longer than two words, respectively. The effect of including a special prediction mechanism of this kind for longer units on prediction accuracy was not tested, however.

Table 2, given as Appendix 1, shows what kinds of theoretical character savings in percentage terms were achieved with our lexicon of multi-word units for the given units by using the best prediction list order. It turned out that the best percentage of character savings was obtained by ordering the prediction list by the length of the unit in terms of the number of word tokens they contain, from the shortest to the longest.

Only some of the matching units found with a given code will be shown to the user on the *prediction list*. Here, the term prediction list means the entire list of multi-word units that are consistent with what the user has typed to the code; the one or ten items that the user actually sees on the list will be a subset of this list. The finding of the length of the unit predicted as the best arrangement principle of the prediction list is consistent with results of prediction experiments for single word prediction reported in the literature, in Swiffin et al. (1987), for instance. Only the results for the best ordering principle of the prediction list will therefore be given in Table 2.

It should be noted that the cost of an extra keystroke for selecting the multi-word prediction mode prior to the prediction of such units is considered in all results of the prediction experiments with multi-word units presented in Table 2. In practice, this cost will lower the percentage of character savings for units predicted somewhat: without this cost, the average percentage of character savings will be seven percent points higher, i.e. 77%, with the same lexicon of multi-word units.

A few statistics are given in Table 2 for the distribution of the results. The most relevant of them are probably the mean percentage of character savings and the median. For each of the 7,009 multi-word units tested, the character savings percentage was calculated individually. As seen, Table 2 contains four columns. The first of them is the name of the statistic. The second one, a perfect user, stands for the prediction method yielding the best percentage of character savings for a perfect user. In practice, the prediction of shorter two-word units is based on the use of two word-initial characters, while that of longer ones on one initial character of some tokens of the unit. The third and fourth columns give the name of the prediction technique by means of which a given unit is predicted; in practice, using either one or two initial characters of some tokens included in the unit.

The main results can be summarized as follows: in the best-case performance, the maximum percentage of character savings that can be achieved for the set of 7,009 multi-word units of legal English as tested here is 70% on average.

4.4 Savings in characters in multi-word prediction with a real text

When evaluating an alternative prediction method, its theoretical prediction accuracy is only one type of evaluation that can be made: we also need to estimate it practical utility with a real text somehow. As mentioned above, the practical utility of multi-word prediction crucially depends on the number of multi-word units that actually appear in a given text, i.e. their *token frequency*.

Erman and Warren (2000: 37), who attempted to quantify the proportion of a sample of text that is accounted for by multi-word-like units, suggest that about 52.3% of the written texts they investigated was made up of pre-fabricated units of varying kinds. Of the nineteen excerpts of texts that they analyzed, between 40% to 60% consisted of ready-made, idiosyncratic combinations of word tokens, that is, of multi-word units of different type. As shown below, a token frequency like that would increase the average percentage of character savings obtained by the two methods of multi-word prediction tested here somewhat over 10% for the whole text.

Based on the analyses of the distribution of the multi-word units in English texts, we can now provide preliminary answers to questions such as the following: What kind of percentage of character savings will be required for the improvement of the prediction accuracy of the whole text in multi-word prediction, say, by 5–10% with respect to the accuracy of traditional prediction methods for single word prediction when predicting the same text with them?

The extent to which multi-word prediction can improve the total percentage of character savings in a hybrid prediction system with a multi-word prediction utility along with single word prediction can be calculated very roughly with the formula given below using the following values:

(1) maximum frequency of occurrence of multi-word units in a given text is 50%. That is, 50% of the tokens of the text appear in multi-word units of some kind;

(2) average percentage of characters saved for the same sequences of words by means of single word prediction is 50%, while in multi-word prediction, the percentage of character savings is 70% on average.

The values above are based on the findings of the token frequency of multi-word units in English texts by Warren and Erman (2000) and accuracy of traditional (*n*-gram-based) prediction methods for single word prediction (about 50% character/keystroke savings), as reported in the literature.

The average percentage of character savings with a mixed prediction scheme of multi-word prediction and single word prediction (word completion) can now be roughly calculated by the following formula:

$$s_c = s_p p + s_w (1 - p),$$

where *p* is percentage of words in multi-word units; s_p is average percentage of character savings in multi-word prediction, and s_w is average percentage of character savings in single word prediction, respectively.

Common values for the performance of the multi-word prediction and methods of single word prediction as tested in this article are s_p = 70% and s_w =50%. The maximum *p* in a given text may be 50%. Thus, we get s_c = 60%. As a result, the multi-word prediction technique can enhance the average character savings by additional 10% for the whole text in comparison to single word prediction under the most favorable conditions, where the coverage of the lexicon is complete for a perfect user who knows the shortest code for the prediction of the multi-word units.

When attempting to evaluate the utility of a prediction method, along with its (theoretical) accuracy in an idealized situation, we should also know how it may perform in practice with a given lexicon and real text. To do that, factors that will undermine the prediction accuracy possible to achieve in theory will have to be considered with respect to the type of savings aimed at (effort savings, time savings, or both) and the physical and visual-cognitive costs of obtaining them in practice. Regarding the latter, metalinguistic skills or memory skills, for example, required in lexical access in multi-word prediction should be considered.

In any kind of prediction system, the lexicon coverage is always incomplete to begin with. As a result, not all units appearing in the text predicted will also be available in the lexicon. If the user tries to expand a non-existing multi-word unit, the cost of the failed prediction may exceed that of predicting the same unit by means of single word prediction, word by word. In this case, the user will have to erase the old code and type in a new one for single word prediction.

It is possible, however, to reduce the cost of failed predictions due to non-existent units in the lexicon by means of interface design: the box where the code is written in the interface of a hybrid prediction system with a multi-word prediction utility can be designed, such that a new code can be typed over the old one directly, as in some Windows applications, without erasing it first. Here, the one keystroke normally spent on erasing the old code will now be saved; saving just one character or keystroke may seem insignificant, but it can actually improve the percentage of characters saved for a given word token or a longer unit more than ten percent points.

Moreover, the number of homographic codes finding both single words and multi-word units when both of them are available in the same lexicon is likely to increase with a realistically large lexicon with (tens of) thousands of items. The rough estimations given in Table 2 will therefore no longer hold. Using larger lexica in prediction affects similarly the accuracy of all prediction techniques, of course, increasing the number of homographic codes, requiring the usage of longer, more distinctive codes in prediction.

Unlike in traditional single word prediction, there is also a need to switch between multi-word prediction and single word prediction in a hybrid prediction system with a multi-word prediction utility according to the appearance of single words and multi-word units in the text predicted. Based on the exact form of the code, however, such a prediction system can also anticipate somewhat the type of unit the user actually is trying to predict and, in these cases, can select an appropriate prediction mode automatically. For example, the code *lete* for *legal tender* would match the onset of no English word, and, if available, only matching multi-word units would be found from the lexicon, while *lem* for *legal matter* would also match the onset of single words such as *lemming*, *lemon*, and *lemur*. The extent to which the selection of an appropriate prediction mode can in practice be done automatically with a given lexicon of multi-word units and single words depends on the number of homographic (overlapping) codes in single word prediction and multi-word prediction, which was preliminarily investigated in Väyrynen et al. (2007).

5. Discussion

In the present study, we have preliminarily estimated the potential of multi-word prediction for improving performance in English word prediction as the average percentage of character savings in the best-case performance. For our purposes, this is the most natural metric – despite its shortcomings – because we want to estimate the maximum utility of multi-word prediction under the most favorable conditions first before evaluating its practical usefulness more thoroughly with real texts and real users in the future.

One important contribution of the present study is how the code length relates to the length of multi-word unit predicted and how a mix of two prediction techniques appears to be the most efficient prediction technique for such units. What is more, the type of code in terms of even or odd number of characters it contains also provides information on what kind of multi-word unit will be predicted (short two-word unit or longer unit), simultaneously maximising the percentage of character savings achievable in multi-word prediction as well.

Somewhat unsurprisingly, the token frequency of multi-word units, not their type frequency, crucially matters in prediction. As shown, to be practically useful, the token frequency of such units in the text predicted will have to be 50% for performance to improve by just 10%, i.e. 50% of the words of the text will have to appear in multi-word units of some type. On the other hand, as shown by Erman and Warren (2000), token frequencies of multi-word units like that do occur in English texts. As a result, word prediction performance can be improved by means of multi-word prediction, at least somewhat, also in practice.

Along with legal English, multi-word prediction can be useful in other domains of use as well. Results of a few prediction experiments in other special application areas available in the literature appear to confirm this. Eng and Eisner (2004), for instance, found that an automated phrase completion feature improved considerably keystroke savings when generating radiology reports by means of word prediction, with a special prediction mechanism for phrases (cf. Foster et al. 2002).

Despite their practical utility in terms of effort or time savings, ideally both, the *user friendliness* of any prediction method is also important. As Lesher (2002) points out, significant keystroke/character savings can be achieved by using complex coding schemes. Unfortunately, such schemes are often impractical for human use. It is therefore likely that the most efficient prediction techniques may not be the most user-friendly ones. As regards the multi-word prediction strategy proposed here, a key question is the ease with which the user can retrieve a multi-word unit from his or her mental lexicon and formulate an appropriate code for its prediction.

Since so much language consists of multi-word units of some kind, it is likely that some way of accessing them is available in the mind of the user. In practice, usability testing is required to find out how to access them in the most user-friendly way. After all, what it is expected in a prediction system is that it will be a valid help for message composition, (ideally) resulting in both effort and time savings when using it. Increasing the cognitive cost required of users to obtain a little enhancement of keystroke/character savings is dangerous for the acceptation of the prediction method proposed.

Along with effort savings, time savings are also important in text production with the help of word prediction, as mentioned above. As well as using a smaller prediction list, greater time savings can be gained by maximizing the expected savings in multi-word prediction by preferring the prediction of longer units instead of (many) shorter ones (Foster et al. 2002: 149). Unfortunately, as shown above, statistically speaking, the former are fairly infrequent.

The manner in which the predictions are sorted on the prediction list may also be relevant from the viewpoint of the time savings achievable in word prediction, for many the real purpose of word prediction (as it is for the maximal character savings achievable in multi-word prediction). Koester and Levine (1998) suggest that the reading of predicted tokens in sequence is less time consuming when the tokens are alphabetically ordered instead of being arranged by frequencies. One keystroke, of course, will always be spent on selecting the correct multi-word unit from the list no matter how the list is ordered.

As pointed out above, the results of the prediction experiments with multi-word units reported here represent the average percentage of characters saved in the best-case performance for a perfect user. The actual performance of a predictor with real users with real texts will always be lower than its theoretical accuracy, of course. This is partly because the user cannot always know the best possible code for finding a given unit, especially in a large lexicon. However, the length of multi-word unit appears to correlate quite well with the type of code that can be used for its prediction.

A prediction system with a multi-word prediction utility could contain a table similar to Table 1, showing the distribution of different units available in the lexicon with instructions for how to access them in the best possible way in prediction. This way, the percentage of character and/or time savings could be maximized. It should be noted that no statistical measures were used to rank the multi-word units predicted, only experiments with a few ordering principles of the prediction lists were carried out here. As a result, Table 2 only evaluates the theoretical character savings for the given multi-word units.

Given that the results of our prediction experiments with two multi-word prediction techniques are preliminary, possible avenues for further research include the following: (1) extent to which the multi-word prediction mode can be selected automatically based on the form of the code used in a hybrid prediction system with a multi-word prediction utility along with single word prediction; an attempt was made to answer that question in part in Väyrynen et al. (2007) already; (2) quantification of the difference in performance between the *n*-gram-based prediction method for multi-word units and the kind of multi-word prediction proposed here with the same test corpus, (3) issues of user interface and user-friendliness of different multi-word prediction techniques and the visual-cognitive loads involved in their usage with real users and texts.

6. Conclusions

In the present study, the potential of multi-word prediction for improving word prediction performance in English word prediction was investigated in the legal English domain. We showed that the two prediction techniques preliminarily tested here can improve the average percentage of character savings about 20% in comparison to single word prediction for a perfect user in the best-case performance. The character or time savings actually gained in practice with a real text crucially depend on the length and token frequency of multi-word units which appear in the text predicted and the way predictions are presented on the prediction list. We also commented on what kind of prediction list order can be used in multi-word prediction with respect to the effort and time savings possible to achieve and how the length of predicted units appears to be related to the code lengths used in prediction and how this correlation can be made use of to practical advantage in multi-word prediction.

References

Alm, Norman; Arnott, John L. & Newell, Alan F. (1992) Prediction and conversational momentum in an augmentative communication system. *Communications of the ACM* 35 (5): 46–57.

Biber, Douglas; Johansson, Stig; Leech, Geoffrey; Conrad, Susan & Finegan, Edward (1999) *Longman Grammar of Spoken and Written English*. Harlow: Longman.

Carlberger, Johan (1998) *Design and Implementation of a Probabilistic Word Prediction Program*. Unpublished Master’s Thesis. Stockholm: Kungliga Tekniska Högskolan.

Chen, Stanley F. & Goodman, Joshua (1996) An empirical study of smoothing techniques for language models. In *Proceedings of the 34th Annual Meeting of the ACL (Santa Cruz, California)*, pp. 310–318.

Cook, Albert M. & Hussey, Susan M. (1995) *Assistive Technologies: Principles and Practice*. Mosby-Year Book Inc.

Eng, John & Eisner, Jason M. (2004) Radiology report entry with automatic phrase completion driven by language modeling. *Radio Graphics* 24: 1493–1501.

Erman, Britt & Warren, Beatrice (2000) The idiom principle and the open choice principle. *Text* 20(1): 29–62.

Foster, George; Langlais, Philippe & Lapalme, Guy (2002) User-friendly text prediction for translators. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002, Philadelphia July 2002)*, pp. 148–155.

Garay-Vitoria, Nestor & Abascal, Julio (2006) Text prediction systems: A survey. *Universal Access in the Information Society* 4: 188–203.

Jackendoff, Ray (1997) *The Architecture of Language Faculty*. Cambridge, MA: MIT Press.

Joutsen, Matti (1985) *English-Finnish Law Dictionary*. Helsinki: WSOY.

Koester, Heidi & Levine, Simon P. (1998) Effect of a word prediction feature on user performance. *Augmentative and Alternative Communication* 14: 25–35.

Langlais, Philippe; Foster, George & Lapalme, Guy (2000) Unit completion for a computer-aided translation typing system. *Machine Translation* 15(4): 267–294.

———— (2002) TRANSTYPE: Development-evaluation cycles to boost translator's productivity. *Machine Translation* 17(2): 77–98.

Lesher, Gregory W.; Moulton, Bryan J.; Higginbotham, Jeffery D. & Alsofrom, Brenna (2002) Limits of human word prediction performance. In *Proceedings of Technology and Persons with Disabilities Conference 2002 (California State University, Northridge)*, pp. 1–4.

Matiasek, Johannes; Baroni, Marco & Trost, Harald (2002) FASTY – A multi-lingual approach to text prediction. In Klaus Miesenberger, Joachim Klaus & Wolfgang Zagler (eds.), *Computers Helping People with Special Needs: 8th International ICCHP Conference (Linz, July 2002)*, pp. 243–250. Berlin/Heidelberg/New York: Springer.

McCoy, Kathleen F. & Demasco, Patrick (1995) Some applications of natural language processing to the field of augmentative and alternative communication. In *Proceedings of the IJCAI '95 Workshop on Developing AI Applications for Disabled People (Montreal, Canada)*, pp. 97–112.

Nantais, Tom; Shein, Frasier & Johansson, Mattias (2001) Efficacy of the word prediction algorithm in WordQ. In *Proceedings of the RESNA 2001 Annual Conference: The Structure of Phonological Representations (Reno, Nevada, June 2001)*, pp. 77–79.

Palazuelos-Cagigas Sira E. (2001) *Contribution to Word Prediction in Spanish and its Integration into Technical Aids for People with Physical Disabilities*. Unpublished Doctoral Thesis. Madrid: Universidad Politécnica de Madrid, Laborotorio de Technologías de Rehabilitación, Dpto. de Ingeniería Elecctrónica.

Palazuelos-Cagigas, Sira E.; Aguilera-Navarro, Santiago; Rodrigo-Mateos, José. L.; Godino-Llorente, Juan. I. & Martin-Sánchez, José L. (1999) Considerations on the automatic evaluation of word prediction systems. In Filip T. Loncke, John Clibbens, Helen H. Arvidson & Lyle L. Lloyd (eds.), *Augmentative and Alternative Communication: New Directions in Research and Practice,* pp. 92–104. London: Whurr.

Palmer, David D. (2000) Tokenisation and sentence segmentation. In Robert Dale, Herman Moisl & Harold Somers (eds.), *Handbook of Natural Language Processing,* pp. 11–35. New York/Basel: Marcel Dekker.

Rosenfeld, Ronald (1996) A maximum entropy approach to adaptive statistical language modeling. *Computer Speech and Language* 10: 187–228.

———— (2000) Two decades of statistical language modeling: Where do we go from here? *Proceedings of the IEEE 88(8) (August 2000)*: 1270–1278.

Swiffin, Andrew L.; Arnott, John L. & Newell, Alan. F. (1987) The use of syntax in a predictive communication aid for the physically handicapped. In *Proceedings of the Tenth Annual Conference on Rehabilitation Technology*, pp. 124–126. Washington, D.C.: RESNA Press.

Villavicencio, Aline; Bond, Francis; Korhonen, Anna & McCathy, Diana (2005) Introduction to the special issue on multiword expressions: Having a crack at a hard nut. *Computer Speech & Language* 19: 365–377.

Väyrynen, Pertti (2005) *Perspectives on the Utility of Linguistic Knowledge in English Word Prediction*. Acta Universitatis Ouluensis. Doctoral Dissertation. Oulu: University of Oulu. http://herkules.oulu.fi/issn03553205/.

Väyrynen, Pertti; Noponen, Kai & Seppänen, Tapio (2007) Analysing performance in a word prediction system with multiple prediction methods. *Computer Speech and Language* 21: 479–491.

Wood, Matthew Edward (1996) *Syntactic Pre-Processing in Single-Word Prediction for Disabled People.* University of Bristol: Unpublished Doctoral Dissertation.

Wester, Malin (2003) *User Evaluation of a Word Prediction System*. Uppsala University, Department of Linguistics: Master's Thesis.

Appendix

Table 2. Percentages of characters saved by ordering the multi-word units on the basis of the number of words separated by a blank, from the shortest to the longest.

Statistic **Min>max**	**Perfect user**	**One word-initial character**	**Two word-initial characters**
One multi-word unit			
Mdn	57.1%	.0%	57.1%
M	42.8%	13.6%	41.1%
Average deviation	29.7%	22.3%	28.6%
SD	32.1%	29.1%	30.9%
Number of multi-word units not predicted	2,416	5,746	2,426
Five multi-word units			
Mdn	68.8%	.0%	64.7%
M	64.9%	31.5%	61.0%
Average deviation	11.4%	36.5%	10.9%
SD	17.8%	37.2%	17.1%
Number of multi-word units not predicted	363	4,061	380
Ten multi-word units			
Mdn	72.2%	69.2%	66.7%
M	70.0%	44.8%	64.4%
Average deviation	8.1%	36.1%	8.7%
SD	12.4%	37.1%	13.0%
Number of multi-word units not predicted	109	2,823	141

Contact information:

Pertti A. Väyrynen
Department of Electrical and Information Engineering
Computer Engineering Laboratory
P.O. Box 4500
90014 University of Oulu
Finland
e-mail: pav(at)ee(dot)oulu(dot)fi