

# **A Memory Aid for Reduced Cognitive Load in Manually Entered Online Bank Transactions**

Frode Eika Sandnes

Faculty of Technology, Art and Design  
Oslo and Akershus University College of Applied Sciences

## **Abstract**

Paper based invoices are still commonly used in Norway where users manually copy large amounts of digits in order to set up transactions. Such digit copying tasks are cognitively and visually demanding, errors are easily introduced and the process is often perceived as laborious. This study proposes an alternative scheme based on dictionary coding that reduces the cognitive load on the user by a factor of five. The strategy has improved error detection characteristics and the scheme is easy to implement.

## **1 Introduction**

In Norway online banking has nearly replaced manual banking. Despite the convenience of direct debit and electronic invoices paper-based invoiced and e-mail invoices are still commonly used. Paper based invoices require the recipient to manually set up the transaction, which involves manually copying up to 50 digits.

The digit copying task is problematic for several reasons. First, copying digits is cognitively demanding. Humans' short term memory is usually limited to  $7\pm 2$  pieces of information at one time [1]. The task therefore has to be split up into several read-input cycles. Given 50 digits up to 10 such read-input cycles are needed, each cycle requiring shifting the visual attention from the source invoice to the transaction input form. Complicating matters further, the invoice identification code is usually the longest sequence of digits and is always presented without separators. The lack of chunking means that additional cognitive and visual effort is required for users to remember and locate where in the string they were when shifting attention between the invoice and the form. Another issue is that email invoices and forms used by the online banks do not follow the Norwegian invoice layout convention and often there is a mismatch in both information sequence and information position making it harder for users to pair invoice data with the required input form fields.

Second, the digit copying task is error-prone. Only the account number and the invoice identification codes utilize simple error detection mechanisms and earlier internet banking implementations did not implement the error correction schemes leading to severe problems [2].

---

*This paper was presented at the NIK-2012 conference; see <http://www.nik.no/>.*

Third, the digit copying task is perceived as time-consuming and laborious. Many users own inexpensive portable laptop computers without a numeric keypad. It has been demonstrated that digit input without numeric keypads are significantly slower than with numeric keypads [3].

Approximately 10% of the population suffers from some form of reduced cognitive functioning or learning disorders such as dyslexia due to reduced memory capacity and effort should be made to reduce the cognitive load on users.

## **2 Invoice structure**

Norwegian invoices have a standardized structure comprising an 11-digit account number, a date, amount and the invoice identification code (KID) with 2 to 25 digits. The last digit in the account number is a modulus-11 error detection code. In this study the bank account parity digit is discarded and an error detection mechanism that applies to the entire transaction is used instead. The last digit of the KID is an error detection code based on modulus-10 or modulus-11. Since KIDs can use one of two error detection schemes the full KID is included in subsequent discussions.

The date indicates the day, month and year of the due date. In this discussion a scheme based on five digits is used comprising two digits for the day of the month, two digits for the month, and one digit representing the least significant digit of the year.

The amount is specified in terms of kroner and øre. In this study the øre part is discarded and amounts are rounded up to the nearest krone since øre no longer is legal tender in Norway. In this study we assume a maximum transaction amount of 99,999 NOK requiring 5 digits.

## **3 Memory aids**

According to cognitive psychology it takes practice to move information from working memory into long term memory. The brain filters and forgets unimportant information and information that we are repeatedly exposed to are gradually fixed to our long term memory such as our personal ID-number. One memory aid employed by many students involves remembering number sequences required to enter a word on a mobile keypad such as a four letter word representing a year in history. Instead of recalling the individual digits where each digit counts as one piece of information, the student remembers the word which counts as one piece of information.

This paper proposes to use a fixed dictionary on the encoding and decoding sides. The principle is to split the sequences of digits into chunks, where each chunk is converted to a linguistic word. Then, the user is presented with a word sequence instead of a number sequence and will thus be able to copy more information per copy-input cycle. For example, take the number sequence 01234 56789. It is first split into chunks of 5 digits namely 01234 and 56789. Each number is then looked up in the wordlist. The digits 01234 could correspond to the Norwegian word 'våg' (dare) and the digits 56789 correspond to the word 'porsjoner' meaning portions. Thus, the word sequence 'våg porsjoner' is presented to the user, who is able to hold both words in working memory and copy the words into the form. Next, the words identified in the wordlist on the receiving end and the indices of the two words are found.

Another advantage of words compared to digit sequences are that experienced readers do not read words character by character, but a word at a time, while digit sequences are

read a digit at a time. The reader recognizes the height signature of text set in lower case letters. This greatly adds to the reading speed.

## 4 Error detection and correction

This strategy proposes three levels of error detection. The first level of error detection catches misspelled words [4] not found in the wordlist. A second level of error detection is achieved by introducing a modulus 10 parity check [5, 6]. This parity check is performed by summing each 5-digit chunk and computing modulus 10 of the total. If there are any mistakes in one or several of the words then this is likely to be caught since the parity will not match. For example, the misspelled words 'urt' and 'tur' that both are valid words in the dictionary are represented by the digits 01196 and 01155, respectively. This difference would flag a modulus-10 parity error.

The third level of error is introduced by taking modulo 2 of each number chunk and representing the result as a bit pattern with the leftmost word representing the least significant bit, etc. This will allow the location of the error to be found in 50% of the cases. More elaborate error detection schemes could be employed. However, optimal codes rely on intricate implementations. Note that no attempt is made correcting errors in our approach. The philosophy is that manual intervention is needed if errors occur to avoid risking erroneous automatic corrections that can lead to financial risks for the user.

The linguistic representation of digits can also assist detecting mistakes when comparing numbers. For instance imagine a customer comparing the two dates 23-06-2012 and 23-08-2012. Each date comprises three information parts, day, month and year. If users focus on the day they might overlook the difference in month since the shape of 6 is similar to 8. However, when comparing the linguistic representations it is obvious that the dates are different.

## 5 Implementation

A Norwegian wordlist found in most Linux distributions containing 181,731 entries with lengths ranging from 2 to 33 characters were used. The entries were sorted according to increasing word length and the 100,000 shortest words were kept. The average length is 8.4 with the longest words being 12 characters long. A proof of concept coder and decoder were implemented in a Microsoft Excel workbook<sup>1</sup>. Excel was chosen in order to prove the simplicity of the approach.

The implementation is demonstrated using an invoice issued by NRK sent to each household with a television:

Accout no.	70410542100
Date	31 Jul, 2012
Amount	1290,06
KID	500330570102919

This is first converted to the digit sequence

01206 70410 54210 31072 01291 50033 05701 02919

which would have to be copied in eight cycles if done manually. This sequence gives the following eight Norwegian words

---

<sup>1</sup>The encoder and decoder prototype can be downloaded from <http://home.hio.no/frodes/KID-DNSv2.3.xlsx>

vas kuwaiterne logopeden tomtens Acta fjernelse dytte lise

This sequence is easily remembered in two chunks, that is ‘vas kuwaiterne logopeden’ and ‘tomtens Acta fjernelse dytte lise’. When this phrase is input to the decoder the original invoice information is detected successfully.

Imagine that one of the words is incorrectly spelled, for instance ‘vkuwaiterne’ instead of ‘kuwaiterne’. The mistake and the location of this mistake would easily be detected as this word is not present in the dictionary. Next, imagine we misspell ‘dytte’ as the valid word ‘dytta’. This mistake is both detected and located since both the modulus-10 and modulus-2 tests fire. Instead, imagine the word ‘dytte’ is misspelled as ‘dytter’. This mistake is detected as the modulus-10 test fires, while the location is not detected since the modulus 2 test does not fire. Finally, imagine that we misspell ‘dytte’ as the valid word ‘dyttet’ — this mistake is not detected with the following setup as the parity tests pass despite the error.

The error detection capability could be drastically improved if the modulus-10 test is replaced with a modulus-100,000 test. Such tests will reduce the chances of missing errors from approximately 10% to 0.001%. These tests require 5 parity digits captured in one linguistic word.

## 6 Conclusions

A memory aid for simplifying the task of manually setting up bank transactions through digit copy tasks is proposed. The strategy converts the digits sequences to sequence of linguistic words allowing transactions to be more easily remembered and compared. The approach has better error-detection capabilities than the digit-based counterpart.

## References

- [1] Miller, G. A. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review* 63, 81-97 (1956).
- [2] Olsen, K. Customer Errors in Internet Banking. *in Proceedings of Norsk Informatikkonferanse 2008*, Trondheim: Tapir Akademisk Forlag (2008).
- [3] Sandnes, F. E. Effects of common keyboard layouts on physical effort: Implications for kiosks and Internet banking. *in Proceedings of Unitech 2010*, Trondheim: Tapir Akademisk Forlag, pp. 91-100 (2010).
- [4] Kukich, K. Techniques for automatically correcting words in text. *ACM Computing Surveys* 24(4), 377-437 (1992).
- [5] Wagner, N. R. and Putter, P. S. Error Detection Decimal Digits. *Communications of the ACM* 32(1), 106-110 (1989).
- [6] Gallian, J. A. Error detection methods. *ACM Computing Surveys* 28(3), 504-517 (1996).