Evaluating a weakness of the National Health Index identifier check-digit to transcription errors in data entry

Jayden MacRae
Patients First
Level 3, 88 The Terrace, Wellington, New Zealand
jayden.macrae@patientsfirst.org.nz

Abstract

We present a simulation model that considers the nature of the NHI identifier check-digit algorithm and its relation to data entry errors. We identify a potential weakness of the algorithm in the detection of transcription errors which is exacerbated by the layout of QWERTY keyboards. We postulate that data entry of NHI identifiers results in the order of 2 in 1000 NHI identifiers entered having a transcription error that goes undetected by the check-digit algorithm. This has implications for those using data sets that contain manually entered NHIs particularly those that are not front-line such as for claim payment or research purposes. This supports the further integration of electronic data interchange between systems to reduce the need for manual data entry and chance of associated errors. We present an alternative approach to calculating a check-digit that would remove the inherent weakness of the existing algorithm by using a modulus-24 or modulus-23 calculation in place of the existing modulus-11.

Keywords: Unique identifier, data entry error

1. Introduction

The National Health Index (NHI) number has underpinned health information in New Zealand for the past 20 years [1]. It is used as a unique identifier in both paper based and electronic systems across all health care settings.

An NHI number has a valid format of seven characters, the first three being alphabetic, the subsequent four being single numeric digits only. The seventh character acts as a check-digit with the purpose of reducing the chances of data entry errors [2]. The NHI uses all English alphabetic characters excluding the letters I and O. Characters are almost always represented in their upper case format. Any single digit number is valid in positions 3-7. This format is shown in Figure 1.

AAA1111

Figure 1 : NHI identifier format where 'A' represents a valid alphabetic character and '1' represents a valid numeric character.

Although the term "NHI number" is in common usage within the health system in New Zealand, the term is somewhat misleading because the identifier itself contains not just numbers but alphabetic characters. For this reason, we refer to the NHI number within this paper as the more generic term NHI identifier.

1.1. Check-digit algorithm

The algorithm used to calculate the check-digit is similar to that seen in the ISBN validation scheme [3]–[5]. An initial check-digit is calculated using the formula seen in Figure 2. Alphabetic character values are determined by their ordinal position within the alphabet excluding 'O' and 'I' using a base of 1 resulting in a valid range of values from 1-24. Numeric character values are determined by their face value resulting in a range of values from 0-9.

$$c_s \quad \sum_{i=1}^{6} \left(v_i . (8 \quad i)\right) \bmod 11$$

Figure 2 : NHI identifier checksum (c_s) formula (i = character position, v_i = the numeric value of the character at position i)

Where the checksum has a result of zero, the NHI identifier is considered immediately invalid otherwise the check-digit is calculated by subtracting the checksum from 11 as shown in Figure 3. The NHI is considered valid when the check-digit is the same as the digit in the seventh position of the NHI identifier.

$$c_d \quad 11 \quad c_s$$

Figure 3 : NHI identifier check-digit (c_d) formula (c_s = check-digit value).

The NHI identifier is assigned to individuals in an arbitrary fashion. They contain no inherent link to an individual. The check-digit algorithm checks that the sequence of characters and numbers conforms to the prescribed format and pattern but does not have any ability to check that the individual who which it is being associated is correct.

1.2. Data Entry Errors and Manual Data Entry

From a data accuracy perspective the best way to exchange information between systems is electronically from system to system. Despite having numerous technologies that support the sharing of health information in an electronic fashion there are still systems that rely on manual data entry. Such systems may be used for service claim processing or research and are often not front-line or direct clinical systems.

Two common types of errors that occur in a manual data entry process are transcription and transposition errors. A transcription error occurs when an atomic unit of data entry, usually a character or keystroke is transcribed from one source to another. A transcription error is usually the result of an operator erroneously pressing an incorrect key on a keyboard or other data entry device but may also result from misreading information from its source.

Transposition errors occur when two atomic units of data entry are swapped in their positions with each other. Transposition errors occur when a data entry operator presses all the correct keys during a data entry operation, but may press them out of sequence.

The NHI identifier check-digit is specifically designed to reduce the chance of these types of data entry errors during manual entry [2].

1.3. The Experiment

This paper outlines a simulation model of the NHI identifier, its check-digit algorithm and how it acts to prevent data entry errors. We consider how the algorithm may contain a weakness in the context of the way in which data is entered from computer keyboards and analyse its cause.

2. Method

We generated every combination of the first six characters of an NHI identifier that conformed to the valid format and stored each value within a PostgreSQL relational database. Each identifier was generated in sequence from AAA000 through to ZZZ999. A valid check-digit was calculated for each record or the record flagged as invalid if a valid check-digit could not be generated.

We decomposed each NHI identifier into its constituent components and stored each character in a separate database field so it was easier to analyse individual frequency and characteristics of characters in each position. We used R and RStudio to perform exploratory analysis of the resultant data set. We looked at the total number of possible NHI identifier combinations, the number of valid combinations and the frequency of individual alphabetic characters within each.

We implemented the NHI algorithm in C#. The algorithm was refactored to output the independent components of the calculation so that we could analyse how each character contributed to the overall check-digit calculation.

3. Results

We generated 13,824,000 combinations of possible NHI identifier values without their associated check-digits. We established that there were 12,567,273 NHI identifiers which were valid; the modulus 11 of their weighted sum (Figure 2) was not zero. There was nothing remarkable in the frequency analysis of the individual NHI identifier components with frequencies being practically indifferent for all characters across all NHI identifiers and only those that were valid.

Table 1 shows the way in which characters are grouped together where they resolve to the same modulus. Characters grouped together can be interchanged together in the same position in an NHI and they will result in the same check-digit digit. For example the letter ‘A’ is interchangeable with the letters ‘M’ and ‘Y’ at the same position in an NHI identifier and such an interchange will not be detected by the NHI check-digit.

		Weight	7	6	5
Characters			Modulus-11 Result		
A	M	Y	7	6	5
B	N	Z	3	1	10
C	P		10	7	4
D	Q		6	2	9
E	R		2	8	3
F	S		9	3	8
G	T		5	9	2
H	U		1	4	7
J	V		8	10	1
K	W		4	5	6
L	X		0	0	0

Table 1 : Modulus-11 results from weighted product of each alphabetic character

There are four groupings of character pairs which lay adjacent to each other in the QWERTY layout which resolve to the same check-digit if transcribed at the same position as each other. These character pairs are shown in Figure 3.

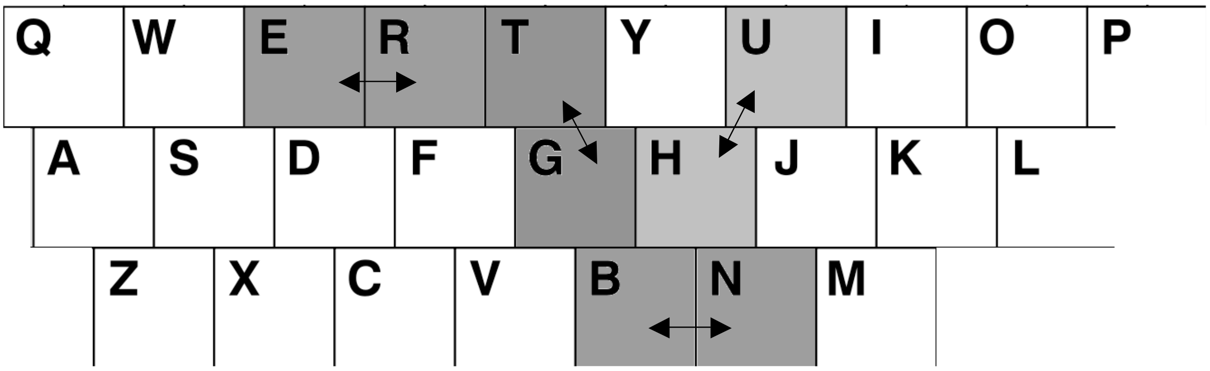

Figure 3 : Character modulus ties on QWERTY keyboard.

The Dvorak keyboard layout has no characters with tied values that are proximal to each other.

4. Discussion

The algorithm used to generate the check-digit for an NHI identifier is very similar to the algorithm used to generate the check-digit for an International Standard Book Number (ISBN). Both schemes use a weighted check-digit approach with a modulus-11 operation. The main difference between the NHI identifier format and the ISBN format is that the NHI identifier uses both alphabetic characters and numeric characters where the ISBN uses only numeric characters. The use of alphabetic characters results in the first three characters of an NHI identifier having one of twenty-four possible values. The result of this is that positions 1-3 within the NHI identifier can have up to three characters that produce an identical check-digit result. For example, an NHI identifier with the character ‘A’ in the first position of the identifier may be replaced with either ‘M’ or ‘Y’ without the check-digit detecting an error. This weakens the utility of the check-digit by decreasing its sensitivity to transcription errors.

The literature on data entry errors suggests a range from 1 to 4%. Expert typists have an approximately 1% error rate per keystroke [6], while standard touch typists may have an error rate at 4% [7]. Each of these error rates are based on the keystrokes used to form natural words. An NHI identifier is not a natural word and therefore any motor learning that may aide the correct keystrokes or keystroke order for typists may not help data entry operators entering NHI identifier type data. Mistyping any of the first three characters would happen in 3% of NHI identifiers entered assuming a conservative error rate of 1% per keystroke. Assuming that each error is independent of the next keystroke, the chance of mistyping 2 or 3 characters in any given NHI identifier are extremely low, being less than 0.1%.

The NHI identifier protects against transposition errors well (refer to Table 1). It is not possible to transpose two characters within the first three without generating a different check-digit because each character value is weighted differently based on its position in the NHI identifier. It may be possible to transpose all three characters in such a way that the check-digit would evaluate correctly but this is likely to be extremely rare given the low chance of a data entry error affecting all three alphabetic characters in a single NHI identifier.

There are eight characters on a QWERTY keyboard that are positioned next to another character that evaluates to the same check-digit. A transcription error caused by a typographical error may be more likely to occur between keys that are adjacent to one another as a typist misses the intended key and strikes a key next to it. The frequency of each character within all possible NHI identifiers is almost identical and therefore for the purpose of this paper we have assumed that any key has a similar probability that it will be involved in an error. Each candidate key has between 3 and 5 other alphabetic keys that lay adjacent to it that may contribute to a transcription error. Based on this we estimate that 8% of transcription errors will result in a transcription that is not detected by the check-digit calculation. This calculation is based on two components; the chance of a key being involved in an error being one of the eight which have a pair with the same check-digit result in an adjacent location on a QWERTY keyboard (33%); and the chance that the key used in error is the pair that shares a check-digit with the target key (25%). The ultimate result of this is that it is possible that there are approximately 2 NHI identifiers with transcription errors in every 1000 which are manually keyed which are likely to evaluate to the same check-digit and go undetected in the absence of other checks.

The layout of alphabetic characters on a Dvorak keyboard is such that no two characters are adjacent to each other which share a common check-digit result in any given position with the NHI identifier. This demonstrates that it is not simply the volume of key pair permutations which results in the situation for the QWERTY keyboard. It would appear to be a coincidence that such an effect has occurred.

Any system that cross references other data with an NHI identifier is far less susceptible to such errors. Typical systems that involve patient interaction usually involve entry of the NHI identifier and further steps to verify a person's identity, such as checking their name, date of birth and address. Not all systems involve direct interactions with patients and therefore may have a lower threshold of identity cross-reference. Such systems may include claim processing systems with manual data entry; such as those still operated by some Primary Health Organisations; or research systems. It is unlikely that erroneous NHI identifier data will come from clinical or patient-facing environments. It is more likely where information is shared in a non-electronic format and re-keying of information into ancillary systems can result in such errors.

There is already strong guidance on the establishment of identity of patients in front-line services to ensure the correct clinical information is delivered and recorded for each individual [8]. We would however recommend that any ancillary systems which obtain NHI identifiers through manual data entry processes employ some type of data cross-reference or check against reference data (such as ensuring a date of birth or surname match an NHI identifier already held on record). The error rate is sufficiently low for us to not recommend a double data entry approach. The Dvorak keyboard layout also protects against the type of transcription error described in this paper but it is likely that the effort and subsequent costs associated with training or retraining data entry operators in using this keyboard layout would far exceed other approaches to mitigate this problem.

If the final check-digit character in the NHI identifier had a range of 24 rather than the existing 10, there would be no instances where two characters in the same position would result in the same check-digit. This could be achieved by using a modulo-24 operation in place of the modulo-11. Twenty-four is not a prime number however and if it was preferable to have the modulus operate on a prime the use of modulo-23 would be a good compromise. This could be achieved by changing the NHI identifier format so that the final check-digit could be represented by an alphabetic character. We note that the newer Health Practitioner Index (HPI) uses an alphabetic character in the check-digit position but still uses a modulus-11 calculation and is so still possibly susceptible to similar transcription errors.

5. Limitations

This paper is largely based on a simulation model to consider how the NHI identifier may be susceptible to a combination of using a shortened check-digit range compared to the range of constituent value inputs and to the configuration of common keyboard layouts. It therefore uses basic assumptions about data entry error rates which have largely been calculated from typists using English words and phrases. How data entry rates of such experiments translate into data entry of sequential identifier data is unclear.

Our calculations of overall NHI identifier errors is based on keystrokes made and they make an assumption that transcription error events are independent of each other. We know that this is unlikely and Matais et al. [7] have identified what they call "chunked" errors where one error is associated with another. The likely impact of this on the entry of NHI identifiers is that we have under-estimated the likelihood of multiple incorrect keystrokes within an NHI identifier. The exact way in which this may affect the entry of NHI identifiers is complex and beyond the scope of this paper.

We also make an assumption that the error rate is the rate at which errors are made in the entry of an NHI but are not detected in themselves (outside of a check-digit). This likely results in an over-estimation of the change of errors overall and errors resulting in unchanged check-digits.

The layout of keyboards may also have an impact on the error rate of data entry across the alphabet. Keys that are to the lateral periphery of the keyboard may be more likely to be mis-keyed. If this was the case, the effect of the QWERTY keyboard layout with several key pairs adjacent to each other may be minimised because they tend to be clustered to the centre of the keyboard. It is difficult to account for individual error frequencies for key position without empirical data.

A more detailed empirical investigation of issues directly related to the keying of identifier type information may provide some of the detail needed to reduce these assumptions.

6. Conclusion

We have undertaken a simulation model to consider the way in which the NHI identifier algorithm operates and how there may be some inherent weaknesses within the algorithm for detecting transcription and transposition errors during manual data entry. We note that the NHI identifier algorithm has an inherent weakness because of its modulus-11 operation allowing more than one character in any given position within the first 3 of an NHI identifier to result in the same check-digit. This allows in some cases for transcription errors to occur without detection by the check-digit algorithm.

We determined that the algorithm performs well to prevent transposition errors mainly due to the way in which each character is weighted differently depending on its position within the NHI identifier.

We found that the layout of the QWERTY keyboard likely further exacerbates the weakness of the algorithm. Eight keys in the keyboard lay adjacent to at least one other key when substitution results in an identical check-digit. This same weakness was not found in the less common Dvorak keyboard layout. We estimate that the rate of NHI identity errors from manual data entry that remain undetected by the check-digit algorithm is in the order of 2 in 1000 NHI identifiers entered.

Our experiment uses a simulation model approach and makes many assumptions. Further research in this area using empirical data will help to clarify the operational implications of our findings.

We suggest an alternate modulus calculation and a check-digit with a wider range of values would likely protect against the weaknesses in the algorithm that we have identified here. Our experiment highlights the prudence in cross-checking additional demographic fields to further establish that NHI identifiers are assigned correct in data sets where they are entered manually and particularly those that are ancillary to front-line services where identity checking may not be done as a matter of course.

7. References

[1] Office of the Privacy Commissioner, "Research Report on the National Health index," Wellington, 2007.

[2] Ministry of Health, "NHI Validation Routine," 2012. [Online]. Available: http://www.health.govt.nz/our-work/health-identity/national-health-index/nhi-information-health-providers/searching-identity. [Accessed: 27-Aug-2015].

[3] J. Gallian and S. Winters, "Modular Arithmetic in the Marketplace," *Am. Math. Mon.*, vol. 95, no. 6, pp. 548–551, 1988.

[4] J. Gallian, "The Mathematics of Identification Numbers.," *Coll. Math. J.*, vol. 22, no. 3, pp. 194–202, 1991.

[5] P. Putter and N. R. Wagner, "Error detecting decimal digits," *Commun. ACM*, vol. 32, no. 1, pp. 106–110, 1989.

[6] J. Grudin, *Error patterns in novice and skilled transcription typing*. New York: Springer, 1983.

[7] E. Matias, M. Scott, and W. Buxton, “One-handed touch typing on a QWERTY keyboard,” *Human-Computer Interact.*, vol. 11, no. 1, pp. 1–27, 1996.

[8] Ministry of Health, “NHI information for health providers: searching identity,” 2015. [Online]. Available: http://www.health.govt.nz/our-work/health-identity/national-health-index/nhi-information-health-providers/searching-identity. [Accessed: 27-Aug-2015].