2016

Enhancing rule-based text classification of neurosurgical notes using filtered feature weight vectors

Sedigheh Khademi
Monash University, sedigh.khademi@gmail.com

Pari Delir Haghighi
Monash University, pari.delir.haghighi@monash.edu

Frada Burstein
Monash University, frada.burstein@monash.edu

Christopher Palmer
Christopher Palmer Ltd

Enhancing rule-based text classification of neurosurgical notes using filtered feature weight vectors

Sedigheh Khademi
Faculty of Information Technology
Monash University
Victoria, Australia
Email: sedigh.khademi@gmail.com

Pari Delir Haghighi
Faculty of Information Technology
Monash University
Victoria, Australia
Email: pari.delirhaghighi@monash.edu

Frada Burstein
Centre for Organisational and Social Informatics
Faculty of Information Technology
Monash University
Victoria, Australia
Email: frada.burstein@sims.monash.edu.au

Christopher Palmer
Christopher Palmer Limited
Email: chris.palmer.nz@gmail.com

Abstract

Clinicians need to record clinical encounters in written or spoken language, not only for its work-flow naturalness but also for its expressivity, precision, and capacity to convey all required information, which codified structured data is incapable of. Therefore, the structured data which is required for aggregation and analysis must be obtained from clinical text as a later step. Specialised areas of medicine use their own clinical language and clinical coding systems, resulting in unique challenges for the extraction process. Rule-based information extraction techniques have been used effectively in commercial systems and are favoured because they are easily understood and controlled. However, there is promising research into the use of machine learning techniques for extracting information, and this research explores the effectiveness of a hybrid rule-based and machine learning-based audit coding system developed for the neurosurgical department of a major trauma hospital.

Keywords

Neurosurgery, Information Extraction, Rule-based expert systems, Machine learning, Audit coding

1 INTRODUCTION

Extracting coded data from clinical notes has received much attention in recent years, as it is imperative that all potential information in electronic health records is made available for data sharing, and for decision support and analytical systems. Whilst computer-based systems are fundamentally structured and their data are available, physicians continue to enter clinical notes in natural language, requiring the expressivity and efficiency of the written or spoken word (Friedman and Elhadad 2014), and so coded data needs to be derived from their notes as a separate step.

Traditionally coded information has been derived from physicians' notes via a manual process of expert human coders reading through notes and deriving codes, and while this is the most accurate approach it is labour intensive and repetitive, expensive, and dependent upon the availability of properly trained personnel. Additionally, despite an expert being the most accurate judge of an author's intentions, they may easily miss information due to inattention.

Research into computer-based solutions for this problem has focussed on two main approaches, one is to devise computer systems which enforce the upfront entry of coded information about the clinical encounter in an unobtrusive way, and the other is to extract coded information after the entry of clinical information into free-text notes (Rosenbloom et al. 2011). For either of these approaches the logic used to suggest codes may be derived by either consultation with a system expert, or by use of computer systems that infer the correct codes via observation of patterns in previously correctly coded data.

To make use of an expert's input for constructing an automated coding system, the expert's knowledge must be the formalised as a series of logical steps – an architecture that is generally described as *rule-based* (Buchanan and Duda 1983). However, text can instead be analysed by computer software for the statistical significance of words and phrases per clinical code found in a large volume of already classified text (a *reference standard*) (Sebastiani 2002), and inferences can be made that certain combinations are most likely indicative of a particular code or codes. This is known as a *machine learning-based* (ML) architecture, where future predictions from new text can be made based on the patterns learned by the software.

Both rule-based and machine learning-based systems have been widely used to automate the extraction of codes from clinical text, and it has been found that the most accurate systems are a hybrid of the two approaches (Minard et al. 2011). The advantage of a rule-based system is that it is does not require a reference standard, is understandable, and open to further refinement to make it increasingly accurate and to cover more text. However, rule-based systems take a lot of consultation and programming effort to create and maintain. A machine learning-based system, being more generally automated, is an easier proposition to create and maintain – but at the expense of requiring a reference standard, and of not being easily understood, and unlike rule-based systems cannot be precisely tuned.

The situation encountered in this research is that of a neurosurgical department of a major trauma hospital, where an internally crafted admissions record system keeps neurosurgical related information in highly abbreviated but jargon-heavy free-text notes, and from which auditing codes are derived in an annual review. The data and the coding systems are highly specific to the department, and we could find no previous research on how to automate the audit code derivation process, a gap this research is devoted to address.

This paper describes how with a review of the literature, and with respect to the requirements of the neurosurgical department, a conclusion was reached that a hybrid approach was optimal. The paper then goes into detail about an aspect of the research, which was how to best enhance the primary rule-based system that was developed with additional machine learning-based predictions, in order to increase the number of useful predictions. The paper describes how these two approaches were integrated by making use of the feature weight vectors of a support vector machine, and how the ML predictions were filtered and refined by post-processing in order to increase their accuracy.

The methodology used in the research is that of *design science*, and the paper summarises the design science approach used, with details of the evaluation and analysis of the final hybrid system. We find that in some cases it is worthwhile incorporating machine learning input, but only after increasing its accuracy by eliminating obviously inappropriate predictions.

2 RESEARCH CONTEXT

The neurosurgical department maintains its own admissions system, which is a simple application that allows the entry of admission records, consisting of diagnostic and procedural codes with an

accompanying note. An individual admission will have one or more admission records (of codes and their accompanying notes), but there is potentially much additional codifiable information in the text of the notes. The department audits the information in the records yearly and a system expert manually reads every note looking for additional un-coded information, as well as to confirm that the originally assigned diagnostic code was accurate. The data derived from this process are in turn codified using codes that are specific to the auditing process, where an auditing code summarizes a number of the more detailed diagnostic codes. The medical terminology used is closely aligned to that published by the Royal College of Surgeons, nevertheless it is highly specific to neurological and neurosurgical conditions, and the abbreviations used can even be specific to an individual practitioner. Table 1 has examples of admission records, their diagnoses and notes, and the related audit codes (data is de-identified).

Admission Code	Diagnosis	Equivalent Audit Code	Notes
0405951142311096	Cranial>Trauma>Extraaxial>SDH	CRANIAL:TRAUMA:SDH	Bilateral small SDH <5mm
0405951142311096	Cranial>Trauma>Intraaxial>SAH (traumatic)	CRANIAL:TRAUMA:SAH	Bilateral frontal and Temporal lobe SAH, R > L
0405951142311096	Cranial>Trauma>Osseous Injury>Skull>Non-displaced	CRANIAL:TRAUMA:SKULL FRACTURE	Non-displaced open # L) parietal bone
00147157907485962	Cranial>Trauma>Intraaxial>SAH (traumatic)	CRANIAL:TRAUMA:SAH	Presumed fall. Multiple bifrontal, Left temporal contusions, SAH, 8mm Left parietal acute SDH
00195986032485962	Cranial>Trauma>Intraaxial>Contusions	CRANIAL:TRAUMA: CONTUSIONS	L cerebellar contusion. Contracoup basifrontal petechial haemorrhages. R SDH.
0570746064186096	Cranial>Trauma>Head Injury, severe	CRANIAL:TRAUMA:TBI	GCS 3. Bilateral fixed and dilated pupils. Occipital #, L frontal SDH, tSAH, contusions
148779928684235	Cranial>Trauma>Extraaxial>EDH	CRANIAL:TRAUMA:EDH	L EDH, contusions, BOS#s, diffuse tSAH, bilat aSDH

Table 1 Admission records, their diagnosis and audit code relationships

The first three records in table 1 are illustrative of a properly coded admission – they are for the same admission but there is one record and accompanying note for each aspect of the patient's condition. The remaining four records are examples where there are many conditions mentioned in the notes, but only one code assigned to the admission, and so there is information in the note that needs to be extracted to derive the extra audit code information. For example the first record of this group should have audit codes CRANIAL:TRAUMA:CONTUSIONS and CRANIAL:TRAUMA:SDH assigned to it in addition to the incoming CRANIAL:TRAUMA:SAH.

3 RELATED WORK

Prior research has described that most electronic health records extensively use narrative text (Häyrinen et al. 2008; Topaz et al. 2016). Entering information as free-form text is the most natural and expressive way for clinicians to record the clinical encounter, however for analysis and re-use of this information, significant clinical information must be extracted from clinical text in a codified format. A variety of technologies have been used to extract coded clinical data via post-hoc text processing, including information extraction, natural language processing (NLP), data mining, and machine learning techniques (Meystre et al. 2008). Most effective recent systems are hybrid which combine technologies such as hand-crafted rules and machine learning (Friedman et al. 2013; Uzuner et al. 2010).

With hybrid systems there are various ways to combine ML and rule-based processes. For example, one strategy is to make use of a decision tree method's output data to derive rules (Farkas and Szarvas 2008), another is to use feature weight data to discover patterns - Guyon et al. (2002) used SVM attribute weights to assist in cancer gene feature selection. Systems developed to extract clinical information from the text of electronic health records range from many that have been focused on concise and often structured documents such as radiography results (Huang et al. 2005) to those that deal with documents having a much higher volume of often more grammatically normal text, such as discharge summaries (Melton and Hripcsak 2005). Some systems have been developed to extract a small component of the text, such as blood pressure results from physicians notes (Turchin et al. 2006), or family history from admission notes (Friedlin and McDonald 2006). If codes are being derived from these, they are usually standard codes such as Unified Medical Language System (UMLS) ("About the UMLS" n.d.) or hospital systems billing codes, as a result there is an expanding resource of NLP systems designed to manage these tasks.

4 RESEARCH DESIGN

4.1 Introduction

The research adheres to Design Science principles, described as a process of creating and evaluating IT artefacts, which must address previously unsolved organizational problems (Hevner et al. 2004), where an *artefact* may be described as a construct, a model, a method, or an instantiation. *Methods* are the processes by which a problem is proposed to be solved, and an *instantiation* is the realization of methods in software or hardware that can be applied to the problem.

The research artefact is a computer-based method designed to generate audit code suggestions based on text classification from notes attached to a record, delivered via a computer programme and report, which will enhance the process of reviewing the notes by a system expert at the time of auditing. The architecture of the method is shown in Figure 1.

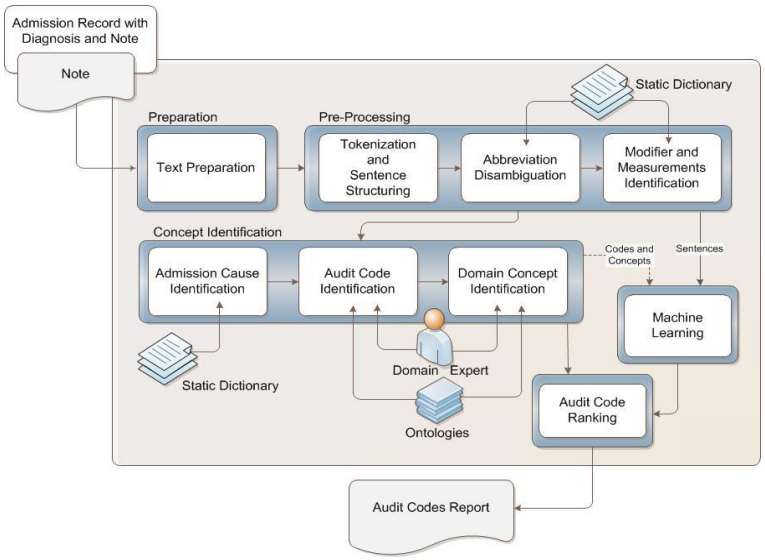

Figure 1 Architecture of the audit code extraction method

After a review of related literature, and assessing the structure of the neurosurgical department's admission notes, together with a requirement for the system expert to be able to understand and control the audit code matching logic, it was decided that a primarily rules-based system would be the best fit. A system using rules can be precisely tuned to suggest the most appropriate audit codes based on the discovery of particular words and phrases, taking into consideration the already existing diagnosis linked to the admission record, and hierarchies of audit code suitability.

However, maintaining rules is an ongoing process, and cannot always account for all of the various phrases used to describe neurological conditions, or even the peculiarities of language used by different practitioners. It was expected that machine learning (ML) could also be utilized to discover statistically significant language patterns, thus increasing the coverage of predictions.

4.2 Rule-based Processing

The rules-based components have been previously described in detail in Khademi et al. (2015). In summary, rules are used throughout the major three stages of the method: Preparation, Pre-processing, and Concept Identification, which culminate in Audit Code generation.

The Preparation stage ensures that the text is suitable for sentence construction through a number of processes that clearly separate punctuation from the text, that disambiguate certain abbreviations, and that correct spelling – these all rely on rules and dictionaries. Pre-processing performs tokenization and sentence structuring, where sentences are derived from notes based on the presence of a comma or full-stop, or the word *and*. Additionally, pre-processing performs lexical tagging and abbreviation disambiguation, which rely on rules encoded as dictionaries. Concept Identification is where words and phrases per sentence are assessed against rules that determine whether they can be classified as being useful for directly suggesting audit codes, or are perhaps additional information such as being about the cause of admission or pertaining to other subject areas such as tests, procedures, and anatomy. The output of the rule-based processing are one or more sentences per note processed, with significant words within the sentence tagged as either audit codes or some other concept.

4.3 Machine learning Processing

The rule-based approach is used as an information extraction process, but needs to be continually refined with additional word combinations. Using Machine Learning techniques to classify text should help uncover previously unknown patterns in the text, and so provide a useful adjunct to an expert's prescriptive rules, leading to further refinements of rules. Additionally, ML classifications might be able to be added into the running application, if text classification can reliably identify audit codes based on statistical probability, without reducing the accuracy of the application. Literature reviews showed that decision tree (Farkas and Szarvas 2008; Huang et al. 2007), probabilistic (Luke Butt, 2013; Pakhomov et al., 2006), and hyperplane classifiers (Aronson et al. 2007; Clark et al. 2008) are widely used for text classification, so a trial was conducted of various ML classifiers from these groups.

The classifiers were trained using the sentences that had been extracted by tokenization, together with the equivalent audit code of the incoming diagnosis. That is, the sentences constitute the data and the codes are the classes required for the ML processing. Although there are some sentences that have no value for predicting audit codes, and some which suggest other codes than the accompanying one, when sufficient good examples exist (i.e. where text matches the accompanying class) then a statistically significant pattern should emerge for reliable predictions.

The data provided by the neurosurgical department had not been supplied as gold-standard training and test sets, and so the approach taken was to use cross validation on all of the data, which consisted of some 12 thousand records covering 66 audit codes, though a small test set was created using a random record selection algorithm to verify the cross validation accuracy. The data was converted into Weka machine learning ARFF (Attribute-Relation File Format) files, using the Weka version 3.7.13 software to process CSV files that had been exported from the original SQL database.

A number of experiments were performed to understand the best performing pre-processing settings, the final choice was to use a String to Word Vector filter, with lower case conversion of words, term frequency–inverse document frequency (TF-IDF) transformation , and with word counts output. There was no stemming performed, and no stop words list was used in the ML data preparation, because the sentences submitted were pre-processed by the rules-based system - which had eliminated stop words. The resulting prepared data is a "bag-of-words" per sentence with each word assigned a numerical score based on its frequency in the sentence and the inverse of its frequency in the entire collection.

TF-IDF transformation gives a lower score to words which appear frequently over the entire collection, so that they do not assume the same significance as more rarely encountered words, but within each document (in this case each sentence) words are then additionally scored based on their frequency (Turney and Pantel 2010). The bag-of-words approach to processing free text takes no account of sentence structure or lexical meaning of words, they are simply evaluated based on their scored frequency. The rules-based system however scores words based on rules about their significance – it looks for specific words as having a positive value, and assigns an even higher value to some of these based on their relationship to the incoming audit code.

Various ML algorithms were assessed, using 10-fold cross validation on the training data, and the three most effective of each of the decision tree, probabilistic, and hyperplane classifiers were J48 Decision Tree (J48), Multinominal Naïve Bayes (NMB), and Sequential Minimal Optimization (SMO) Support Vector Machine. The SMO support vector machine scored best with 63.1% correctly classified (measured using Recall) – see table 2.

	J48	NBM	SVM
Precision	0.579	0.609	0.611
Recall	0.598	0.620	0.631
F-measure	0.577	0.608	0.612

Table 2 Comparative accuracies of three Machine Learning methods

The best performing SMO support vector machine used the Weka defaults of a linear kernel (PolyKernel –E 1.0) and complexity 1 (-C 1.0), and was picked as the most useful of all models tested. Its classification accuracy was higher than the others. It was relatively quick to process, and the text output of running the model provided feature weight vector data that could be integrated back into the application to provide a seamless ML predictive capacity.

4.3.1 Using Feature Weight Vectors

The standard workflow required to make use of the Weka SMO model for predictions on new data is a batch process that cannot be integrated into a running SQL-based application, but it was observed that the data output by the Weka learning process was adaptable for integration. Weka's results output file for the SMO SVM classifier contains pairs of comparisons between every possible class, where a numerical score is given to every word observed for a class, compared with observations of those same words in the class being compared to. These are labelled as "attribute weights" by Weka, but are more generally known as the feature weight vectors of an SVM. In most cases a particular class scores certain words better than for any other class, and these scores can be utilised programmatically to arrive at a most likely class for a sentence. Therefore, the attribute weight data was extracted from the text file (using an R script) and imported into the SQL database, and a SQL function was written to use this data for scoring sentences directly within the application.

This approach proved to duplicate the original scoring by Weka in most cases, certainly reliably enough to allow the integration of the ML predictive process into the SQL application. As a consequence, the application is capable of taking a note as an input, splitting it into sentences, processing them via both the rule and ML-based components, and delivering a list of predictions – all within a split second. The application is capable of integrated real-time rule-based and ML predictions at data entry time, with periodic re-importing of new feature weight vectors required to keep the ML component updated.

4.4 Filtering the Predictions

A final rule-based ranking process assesses the frequency and relevance of the predicted audit codes, to confirm that they either duplicate the incoming audit code, or predict useful additional codes, or are in fact not useful to retain. There is further description of this process in the evaluation section, where its implications for filtering ML predictions is explained in more detail.

4.5 Presenting the Predictions

The refined list of audit codes per note is added to a table which forms the basis of a report – which can be used in summary form for overall counts of additional predictions per incoming audit code. It also allows drill through to a detailed report so that an auditor using the report can confirm predictions by inspecting notes. The reports can be filtered to present subsets of data, based on the incoming audit code and the predicted codes.

5 EVALUATION

5.1 Evaluation Measures

Standard evaluation metrics of Precision, Recall and F-score were employed (Stanfill et al. 2010). Two methods for calculating overall evaluation measures have been used: micro-averaging and macro-averaging, as explained by Manning (Christopher D. Manning, 1999). Micro-averaging computes the summation of all the individual class scores into one contingency table and then the evaluation measures are calculated based on the totals in this table. Macro-averaging calculates recall, precision, and F-measure for each class, and then computes weighted averages of these scores. Micro-averaging gives more importance to the classes with larger number of instances, whereas macro-averaging evaluates the performance of the classifier across all the classes more fairly.

5.2 Evaluation of the Two Approaches

Evaluation of the accuracy of the predictions from the rule-based approach versus the machine learned predictions confirmed that overall a rule-based system was the best choice, given that the rule-based system is more accurate and with better coverage, is understandable, and is readily adjustable. Using macro-weighted averaging over a set of 12,023 admission records the rules-based system was able to achieve an F-score of 0.749, compared with 0.672 from the support vector machine, shown in table 3.

		Rule Based			SVM Based		
	Class Total	Matched	Predicted		Matched	Predicted	
Totals	12023	8159	12023		7631	12023	
		Precision	Recall	F Score	Precision	Recall	F Score
Micro averages		0.679	0.679	0.679	0.635	0.635	0.635
Macro Weighted averages		0.777	0.754	0.749	0.695	0.680	0.672

Table 3 Summary of results of rule-based vs support vector machine-based prediction accuracy

5.3 A Hybrid Approach

In some cases, the ML-based approach makes valid predictions that are missed by the rule-based approach, and with some audit codes the ML-based approach has a higher F-score than the rule-based one, so for these the ML-based predictions should be incorporated into the method. However, the advantages of retaining a primarily rule-based approach are such that it is not optimal to simply replace the rule-based predictions, rather the ML-based predictions should be *combined* with the rule-based ones to obtain an overall better prediction. A hybrid approach is required – retaining the rule-based approach allows for ongoing rules adjustments to improve accuracy and coverage, adding in the ML-based predictions where it improves overall F-score allows for the insights of the statistical approach. To illustrate where a combined approach works, consider the following table of the combined predictions for the COMPLICATION vs. CRANIAL:TRAUMA:SKULL FRACTURE audit codes:

Diagnosis Audit Code	Class Total	Rule Based Matched	Rule Based Predicted	Rule Based Precision	Rule Based Recall	Rule Based FScore	SVM Based Matched	SVM Based Predicted	SVM Based Precision	SVM Based Recall	SVM Based FScore
COMPLICATION	465	144	171	0.842	0.310	0.453	170	224	0.759	0.366	0.494
CRANIAL:TRAUMA: SKULL FRACTURE	369	322	443	0.727	0.873	0.793	327	751	0.435	0.886	0.584

Diagnosis Audit Code continued…	Combined Matched	Combined Predicted	Combined Precision	Combined Recall	Combined FScore	Combined - Rule based Difference
COMPLICATION	223	300	0.743	0.480	0.583	0.130
CRANIAL:TRAUMA: SKULL FRACTURE	352	838	0.420	0.954	0.583	-0.210

Table 4 Combined Prediction Systems

By combining the rule-based and (SMO) SVM-based predictions for COMPLICATION the F-measure is increased by 0.130 from 0.453, to 0.583. Note that there are an increased number of invalid predictions, so that the combined precision drops from 0.842 to 0.743, but the increased recall from 0.310 to 0.480 produces the overall benefit. The effect of this is most easily visualised in a Venn diagram as the combinations of total predictions vs correct predictions (matches) from both sets. Where predictions are correct they are coloured blue and green, and where they overlap (predicting identically) then they are blue-green (that is, the internal ovals and their overlap). Where predictions are incorrect they are coloured red (the external areas), it is obvious that by combining the predictions a greater number of correct predictions are obtained in relation to the increased number of incorrect predictions, with an overall benefit gained.

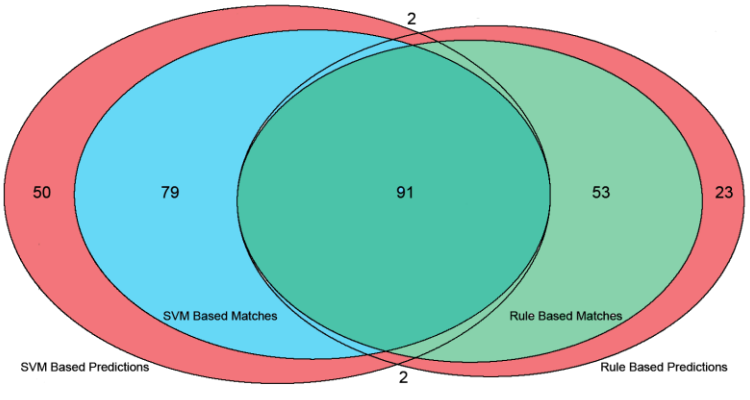

Figure 2 COMPLICATIONS audit code: Combined SVM and rule-based predictions

Almost a third of the audit codes showed an improved F-score when using a combined approach, these tended to be codes where there was a lot of variation of terms used in the notes, which had not yet had a rule created for them.

5.4 Hybrid Approach Results Analysis

The method identifies individual sentences in the admission record's note, with the aim of predicting the most representative audit code per sentence. The expectation is that there is likely to be at least one sentence that will deliver the currently assigned (incoming) audit code, after which any other audit codes

can be assessed for usefulness and likelihood for reporting of additional codes. Codes may come from either the rule-based predictive process or from the ML-based process, table 5 contains examples:

No.	Assigned Audit Code	Notes	Sentences	Rule-Based Prediction	SVM Prediction
1	CRANIAL:TRAUMA:SAH	Transfer from XXXX. Small L tempoparietal EDH, R frontal and temporal contusions, R parietal#, SAH	Transfer from XXXX Small L tempoparietal EDH R frontal temporal contusions R parietal # SAH	(None - eliminated as words are unidentified) CRANIAL:TRAUMA:EDH (see example 2) (None - but retained as words are diagnostic) CRANIAL:TRAUMA:CONTUSIONS (see 3) CRANIAL:TRAUMA:SKULL FRACTURE (4) CRANIAL:TRAUMA:SAH	CRANIAL:TRAUMA:SDH CRANIAL:TRAUMA:SKULL FRACTURE CRANIAL:TRAUMA:CONTUSIONS SPINE:TRAUMA:FRACTURE CRANIAL:TRAUMA:SAH
2	CRANIAL:TRAUMA:EDH	R frontoparietal EDH. Intoxicated fall w HS	R frontoparietal EDH Intoxicated fall w HS	CRANIAL:TRAUMA:EDH (None - but retained as words are traumatic)	CRANIAL:TRAUMA:EDH CRANIAL:TRAUMA:TBI
3	CRANIAL:TRAUMA:CONTUSIONS	Inferior left frontal lobe cerebral contusions and haematoma	Inferior left frontal lobe cerebral contusions haematoma	CRANIAL:TRAUMA:CONTUSIONS CRANIAL:TRAUMA	CRANIAL:TRAUMA COMPLICATION:POSTOP BLEED
4	CRANIAL:TRAUMA:SKULL FRACTURE	Bilateral PTB/clinoid #. Surfing accident. No intracranil haem.	Bilateral PTB/clinoid # Surfing accident No intracranial haem	CRANIAL:TRAUMA:SKULL FRACTURE (None - but retained as words are traumatic) *Negated* CRANIAL:TRAUMA:ICH	CRANIAL:TRAUMA:SKULL FRACTURE CRANIAL:TRAUMA CRANIAL:TRAUMA:SAH
5	CRANIAL:ANEURYSM (UNRUPTURED)	Bilateral MCA aneurysms. ?vasculitis	Bilateral MCA aneurysms ? vasculitis	CRANIAL:ANEURYSM probable COMPLICATION:INFECTION	CRANIAL:ANEURYSM (UNRUPTURED) CRANIAL:OTHER
6	SPINE:DEGENERATIVE	Right L4/5-L5/S1 redo, decompression, discectomies and unilateral TLIF	Right L4/5-L5/S1 redo decompression discectomies unilateral TLIF	SPINE:OTHER	COMPLICATION SPINE:CANAL STENOSIS SPINE:DEGENERATIVE SPINE:OTHER

Table 5 Deriving Audit Codes per Sentence

In table 5, the assigned audit code is the single audit code that has arrived with the incoming note. As described in sections 4.2 and 4.3, sentences are derived from notes, then key words are extracted and classified as audit codes or other relevant concepts via a rule-based information extraction process, with sentences containing useful data forming the input for the ML (SVM) learning process.

In the first example in table 5, no useful words were identified in the first sentence (transfer from another, de-identified, hospital), so it is not passed to the ML process. All the other sentences are used to train (and later get predictions from) the ML process as indicative of the incoming audit code CRANIAL:TRAUMA:SAH. Although the entire note may be considered as indicative of SAH, individual sentences may more clearly point to other audit codes. This is evidenced by the choices made by both the rule-based and SVM-based systems, which predict other codes, only agreeing once with SAH as a code. The accompanying Note example numbers 2 through 4 illustrate how the predictive systems have arrived at their choices for the individual sentences in the first example. That is, example numbers 2 to 4 typify accurate primary predictions for the additional codes that were identified in the first example, the additional codes in example number 1 have been referenced to link them to their respective example numbers 2 through 4.

In example number 1 only one sentence is clearly indicative of SAH, which is the very last single-word sentence of the Note: SAH. In the system's design there is no requirement for reporting the already assigned SAH, so that sentence can be used to confirm an accurate prediction for the record, but apart from that has no purpose. The other sentences that do not agree with the incoming code are candidates for reporting as additional or more correct codes. Even though when compared against the incoming SAH code they are inaccurate, when compared to the information in the individual sentences their predictions may be relevant.

Note that in these first four examples very little benefit is obtained by the ML predictions, and in some cases it predicts where the words in the sentence would only suggest the prediction because of statistical inference – for example "surfing accident" and "intoxicated fall w HS" are not enough for rule-based predictions, whereas the ML-based system predicts CRANIAL:TRAUMA and CRANIAL:TRAUMA:TBI. Example 5 is an instance where the ML-based prediction is more correct than the rule-based one for the type of aneurysm, and it is these kinds of word combinations that a ML-based system can readily discover.

Despite any potential accuracy, not every audit code prediction made by the system should be reported as additional codes. This is because some codes should be considered irrelevant in the context of the incoming audit code. Therefore, a further rule-based filtering process takes place before the additional codes are retained. Example 3 in table 5 illustrates why this is required, the incoming code of CRANIAL:TRAUMA:CONTUSIONS is specific enough not to benefit from additionally reporting the prediction of CRANIAL:TRAUMA, which is a more general code – a more precise code is preferred. Furthermore, this

example also contains a SVM prediction of COMPLICATION:POSTOP BLEED, and this should also be discarded since it is not likely to be relevant in the context of reporting on a major head injury.

Example 6 shows that while the SVM did make a correct match to the incoming code on one of the sentences, filtering is required to eliminate irrelevant predictions. It has been determined that SVM predictions of SPINE:CANAL STENOSIS and SPINE:OTHER are not useful as they do not contribute to an improved F-measure, so these would not in fact be used, leaving just the COMPLICATION code as potentially significant.

Because the rule-based system is inherently understandable and refinable, its predictions are preferred. For instance, any more precise finding for a sentence from the rule-based system is there because important key words have been discovered, therefore the prediction is more reliable. Additionally, the rule-based system's negated predictions should be preferred, example 4 above for the sentence "no intracranial haem" illustrates this.

After filtering has taken place many predicted additional codes are eliminated as either irrelevant when compared to the incoming code, or because they are ML prediction classes that have been determined as not contributing anything useful to the system. Analysis of the remaining codes show that few ML predictions are useable, as by-and-large they have either already been correctly predicted by the rule-based system, or they are irrelevant.

5.5 Expert Evaluation

A domain expert was able to evaluate 100 predictions and make comments. The majority of the predictions were considered as correct, but where there was a suggestion of different or additional codes then these were accompanied by explanations that could lead to refinement of the rules. For instance, on records where multiple head injury-related codes were found but no precise suggestion of a code for traumatic brain injury (TBI), the expert suggested that this should also be added – a rule can easily be added where TBI is included in these contexts. Occasionally the expert suggested refinements of the predictive logic that were beyond the capacity of the system to suggest – they would require more sophisticated natural language processing, which could be incorporated after additional research. Of great significance is that when making his analysis the expert used logical rule-based thinking - "if this then that", and understood the system as something that should work on a rule-based approach and be capable of refinement by using the rule-based approach.

6 CONCLUSION

The research has demonstrated that machine learning-based predictions can add value to a rule-based neurosurgical text classifier, when the ML predictions are utilised to gain extra insight into expressions that are statistically significant, and which have not been previously described by the system expert. Additionally, apart from examining the text that is classified by a machine for insight into possible new rules, where ML predictions consistently predict a class well then its predictions can be integrated into a working application. Further rule-based filtering can remove obviously unsuitable predictions.

It has also been noted that when a system expert is envisaging an ideal system then he tends to describe it as one that follows certain logical rules, and expects to be able to tune it based on refinements of rules. Therefore, a primarily rule-based solution is most likely to satisfy the expert's requirements, despite a machine learning algorithm making similar suggestions to a rule-based system. Even if a primarily machine learning based system was designed, it would still require manual refinement.

7 FUTURE RESEARCH

There is an interesting possibility that the meta data generated by the rule-based system could be combined with the text to provide a greater number of patterns for the machine learning system to learn on. For instance, there are many words that are classified by the rule-based system as anatomical, such as "anatomical:brain", "anatomical:skull", and "anatomical:spine". Currently the rule-based system does not use this data but since these classifications tie together many more complex words, some of which appear very infrequently, it's possible that adding these classifications to the text, and even also the audit codes deduced by the rules-based system, would assist the ML process to be more accurate.

The use of the output of the ML support vector machine to provide a table of word weightings proved to be very useful, it meant that this data could be made use of by the application directly, rather than having to pass processing back to the ML software. There is opportunity to refine the way the ML data is used, especially when it comes to eliminating obvious mistakes by the ML system. The next most accurate ML

algorithm was the Naïve Bayes process, and its output is also amenable to incorporation into the application, so it would be good to be able to test the result of combining these two ML systems.

To deal with class imbalance issues, under and over data sampling techniques could be explored, and other machine learning processes such as cost-sensitive classifiers could also be evaluated. Different ML software are certainly worth consideration – this research used the Weka data mining software but it would be interesting to explore libraries written in R or Python. R is especially an intriguing option as with the introduction of SQL Server 2016 it is now possible to embed R functionality into a SQL Server programme.

8 REFERENCES

"About the UMLS.," (n.d.)., Product, Program, and Project Descriptions, (available at https://www.nlm.nih.gov/research/umls/about_umls.html; retrieved September 30, 2016).

Aronson, A. R., Bodenreider, O., Demner-Fushman, D., Fung, K. W., Lee, V. K., Mork, J. G., Névéol, A., Peters, L., and Rogers, W. J. 2007. "From Indexing the Biomedical Literature to Coding Clinical Text: Experience with MTI and Machine Learning Approaches," in Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language ProcessingBioNLP '07, Stroudsburg, PA, USA: Association for Computational Linguistics, pp. 105–112 (available at http://dl.acm.org/citation.cfm?id=1572392.1572412).

Buchanan, B. G., and Duda, R. O. 1983. "Principles of Rule-Based Expert Systems," in Advances in ComputersAdvances In Computers, M. C. Yovits (ed.) (Vol. 22), Elsevier, pp. 163–216 (available at http://www.sciencedirect.com/science/article/pii/S0065245808601291).

Clark, C., Good, K., Jezierny, L., Macpherson, M., Wilson, B., and Chajewska, U. 2008. "Identifying Smokers with a Medical Extraction System," Journal of the American Medical Informatics Association (15:1), pp. 36–39 (doi: 10.1197/jamia.M2442).

Farkas, R., and Szarvas, G. 2008. "Automatic construction of rule-based ICD-9-CM coding systems," *BMC Bioinformatics* (9:Suppl 3), p. S10 (doi: 10.1186/1471-2105-9-S3-S10).

Friedlin, J., and McDonald, C. J. 2006. "A Natural Language Processing System to Extract and Code Concepts Relating to Congestive Heart Failure from Chest Radiology Reports," AMIA Annual Symposium Proceedings (2006), pp. 269–273.

Friedman, C., and Elhadad, N. 2014. "Natural Language Processing in Health Care and Biomedicine," in Biomedical InformaticsE. H. Shortliffe and J. J. Cimino (eds.), Springer London, pp. 255–284 (available at http://link.springer.com/chapter/10.1007/978-1-4471-4474-8_8).

Friedman, C., Rindflesch, T. C., and Corn, M. 2013. "Natural language processing: State of the art and prospects for significant progress, a workshop sponsored by the National Library of Medicine," Journal of Biomedical Informatics (46:5), pp. 765–773 (doi: 10.1016/j.jbi.2013.06.004).

Guyon, I., Weston, J., Barnhill, S., Vapnik, V., 2002. Gene Selection for Cancer Classification using Support Vector Machines. Mach. Learn. 46, 389–422. doi:10.1023/A:1012487302797

Häyrinen, K., Saranto, K., and Nykänen, P. 2008. "Definition, structure, content, use and impacts of electronic health records: A review of the research literature," International Journal of Medical Informatics (77:5), pp. 291–304 (doi: 10.1016/j.ijmedinf.2007.09.001).

Hevner, A. R., March, S. T., Park, J., and Ram, S. 2004. "Design Science in Information Systems Research," MIS Q. (28:1), pp. 75–105.

Huang, Y., Lowe, H. J., Klein, D., and Cucina, R. J. 2005. "Improved Identification of Noun Phrases in Clinical Radiology Reports Using a High-Performance Statistical Natural Language Parser Augmented with the UMLS Specialist Lexicon," Journal of the American Medical Informatics Association : JAMIA (12:3), pp. 275–285 (doi: 10.1197/jamia.M1695).

Huang, Y., McCullagh, P., Black, N., and Harper, R. 2007. "Feature selection and classification model construction on type 2 diabetic patients' data," Artificial Intelligence in Medicine (41:3), pp. 251–262 (doi: 10.1016/j.artmed.2007.07.002).

Khademi, S., Haghighi, P. D., Lewis, P., Burstein, F., and Palmer, C. 2015. "Intelligent audit code generation from free text in the context of neurosurgery," (available at https://www.researchgate.net/profile/Frada_Burstein/publication/286590834_Intelligent_au

dit_code_generation_from_free_text_in_the_context_of_neurosurgery/links/56a7364408ae860e025548ba.pdf).

Luke Butt, G. Z. 2013. "Classification of cancer-related death certificates using machine learning," The Australasian medical journal (6:5), pp. 292–9 (doi: 10.4066/AMJ.2013.1654).

Manning Christopher D. 1999. Foundations of statistical natural language processing, Cambridge, Mass.: MIT Press

Melton, G. B., and Hripcsak, G. 2005. "Automated Detection of Adverse Events Using Natural Language Processing of Discharge Summaries," Journal of the American Medical Informatics Association : JAMIA (12:4), pp. 448–457 (doi: 10.1197/jamia.M1794).

Meystre, S., Savova, G., Kipper-Schuler, K., and Hurdle, J. 2008. "Extracting information from textual documents in the electronic health record: a review of recent research.," Yearbook of medical informatics, pp. 128–144.

Minard, A.-L., Ligozat, A.-L., Ben Abacha, A., Bernhard, D., Cartoni, B., Deléger, L., Grau, B., Rosset, S., Zweigenbaum, P., and Grouin, C. 2011. "Hybrid methods for improving information access in clinical documents: concept, assertion, and relation identification," Journal of the American Medical Informatics Association : JAMIA (18:5), pp. 588–593 (doi: 10.1136/amiajnl-2011-000154).

Pakhomov, S. V. S., Buntrock, J. D., and Chute, C. G. 2006. "Automating the Assignment of Diagnosis Codes to Patient Encounters Using Example-based and Machine Learning Techniques," Journal of the American Medical Informatics Association : JAMIA (13:5), pp. 516–525 (doi: 10.1197/jamia.M2077).

Rosenbloom, S. T., Denny, J. C., Xu, H., Lorenzi, N., Stead, W. W., and Johnson, K. B. 2011. "Data from clinical notes: a perspective on the tension between structure and flexible documentation," Journal of the American Medical Informatics Association : JAMIA (18:2), pp. 181–186 (doi: 10.1136/jamia.2010.007237).

Sebastiani, F. 2002. "Machine Learning in Automated Text Categorization," ACM Comput. Surv. (34:1), pp. 1–47 (doi: 10.1145/505282.505283).

Stanfill, M. H., Williams, M., Fenton, S. H., Jenders, R. A., and Hersh, W. R. 2010. "A systematic literature review of automated clinical coding and classification systems," Journal of the American Medical Informatics Association : JAMIA (17:6), pp. 646–651 (doi: 10.1136/jamia.2009.001024).

Topaz, M., Lai, K., Dowding, D., Lei, V. J., Zisberg, A., Bowles, K. H., and Zhou, L. 2016. "Automated identification of wound information in clinical notes of patients with heart diseases: Developing and validating a natural language processing application," International Journal of Nursing Studies (64), pp. 25–31 (doi: 10.1016/j.ijnurstu.2016.09.013).

Turchin, A., Kolatkar, N. S., Grant, R. W., Makhni, E. C., Pendergrass, M. L., and Einbinder, J. S. 2006. "Using Regular Expressions to Abstract Blood Pressure and Treatment Intensification Information from the Text of Physician Notes," Journal of the American Medical Informatics Association : JAMIA (13:6), pp. 691–695 (doi: 10.1197/jamia.M2078).

Turney, P. D., and Pantel, P. 2010. "From Frequency to Meaning: Vector Space Models of Semantics," J. Artif. Int. Res. (37:1), pp. 141–188.

Uzuner, Ö., Solti, I., and Cadag, E. 2010. "Extracting medication information from clinical text," Journal of the American Medical Informatics Association : JAMIA (17:5), pp. 514–518 (doi: 10.1136/jamia.2010.003947).

ACKNOWLEDGEMENTS

Dr Phil Lewis, Senior Research Fellow, Monash University, had the original inspiration for the research, and throughout has kindly provided neurosurgical expertise, guidance, and evaluation of the project.