Automatic Methods for Coding Historical Occupation Descriptions to Standard Classifications

Graham Kirby
School of Computer Science
University of St Andrews
Fife KY16 9SX, Scotland
gnck@st-andrews.ac.uk

Jamie Carson
School of Computer Science
University of St Andrews
Fife KY16 9SX, Scotland
jkc25@st-andrews.ac.uk

Fraser Dunlop
School of Computer Science
University of St Andrews
Fife KY16 9SX, Scotland
frjd2@st-andrews.ac.uk

Chris Dibben
Longitudinal Studies Centre Scotland
Universities of St Andrews & Edinburgh
cjld@st-andrews.ac.uk

Alan Dearle
School of Computer Science
University of St Andrews
Fife KY16 9SX, Scotland
alan.dearle@st-andrews.ac.uk

Lee Williamson
Longitudinal Studies Centre Scotland
Universities of St Andrews & Edinburgh
lepw@st-andrews.ac.uk

Eilidh Garrett
Department of Geography
University of Cambridge
eilidh.garrett@btinternet.com

Alice Reid
Department of Geography
University of Cambridge
alice.reid@geog.cam.ac.uk

Abstract

The increasing availability of digitised registration records presents a significant opportunity for research in many fields including those of human geography, genealogy and medicine. Re-examining original records allows researchers to study relationships between factors such as occupation, cause of death, illness, and geographic region. This can be facilitated by coding these factors to standard classifications. This paper describes work to develop a method for automatically coding the occupations from 29 million Scottish birth, death and marriage records, containing around 50 million occupation descriptions, to standard classifications. A range of approaches using text processing and supervised machine learning is evaluated, achieving accuracy of 92.3 ± 0.2% on a smaller test set. The paper speculates on further development that may be needed for classification of the full data set.

1 Introduction

This paper describes work being carried out to develop a method for automatically processing the 50 million occupations recorded on Scottish birth, death and marriage records from 1855 to the present day.

There are two key problems: firstly, how to consistently code occupations over the entire 150 year period so that researchers can explore changing patterns and trends; and secondly, how to automate this process so that the majority of records do not need to be manually coded. In this paper we focus on the second of these problems: developing methods to automatically classify narrative occupation descriptions into a fixed set of standard classifications.

This work builds on previous efforts to automatically classify causes of death contained in the Scottish records mentioned above (Carson et al., 2013).

2 Data Set

The target data set is the entirety of births, deaths and marriages recorded in Scotland from 1855 to present day, comprising 29 million events and around 50 million occupations. These records will ultimately be coded to the HISCO classification (van Leeuwen, Maas, and Miles, 2002; HISCO, 2013).

In order to develop our approach and to trial various methodologies, we have conducted experiments on a smaller data set, originating from the Cambridge Family History Project (Bottero and Prandy, 2001), cited by Prandy (2012). This set contains 243,000 records dating from early C18th to present day, with 30,200 unique occupation descriptions. The data was completely anonymised before processing in our classification experiments.

All of the records were previously classified to SOCH, an extension of the SOC coding system (Bureau of Labor Statistics, 2010) defined in the Family History Project to include historical terms. SOCH includes 1,000 distinct codes, of which 453 occurred in the data set. A subset of the data, comprising 64,000 of the records with 9,400 unique occupation descriptions, was also classified to HISCO, via a fixed SOCH-HISCO mapping. The HISCO classification includes 1,675 codes, of which 337 occurred in the data subset. This subset of the data was used in the classification experiments.

3 Approaches to Classification

Based on previous experience with automatic classification of causes of death (Carson et al., 2013), a number of classification techniques were evaluated, and their results compared with the assumed 'gold standard' of the domain expert coding. The following aspects were investigated:

- the effects of various types of data cleaning
- the accuracy of three different automatic classifiers
- the accuracy of three different ensemble classifiers
- the effects of preparatory feature selection
- the effects of relaxing the correctness condition in measuring classification accuracy

3.1 Cleaning

The previous work on classification of causes of death involved textual descriptions written by doctors and other officials. These were often verbose, containing various filler words and parenthetical comments. In that work, it proved useful to employ a number of simple cleaning rules using a regular expression processor, to remove some of these non-relevant words, and to expand commonly used abbreviations. For example:

Original Text	Cleaned Text
(cardiac paralysis) diphtheria	diphtheria
1 paralysis 2 smallpx	paralysis
both flu; acute pneu	influenza
injury caused by being run over by a railway truck	injury

Table 1. Example cleaning of cause of death strings.

In the current work on classifying occupations, after studying example phrases, it was concluded that this type of cleaning would not be necessary, since most occupation descriptions only contain a few words. For example:

- wharfingers clerk
- wheat bag stacker
- wheeler
- wheelright
- wheelsmith (railway works)
- wheelwright

The consistency of human coders was also considered. The HISCO-coded data contained 9,400 unique occupation descriptions. 273 occurred at least 25 times. 119 had been multiply-coded to more than one HISCO code.

Of the multiply-coded descriptions, there were only 10 which had more than 5% alternative codings. The most commonly alternatively coded descriptions are shown below, with the most frequent codings for each:

- fireman
  - railway steam-engine fireman (37%)
  - fire-fighters (26%)
  - boiler fireman (13%)
- machinist
  - sewers and embroiderers (53%)
  - machine-tool operators (46%)
  - printing pressmen (1%)

- iron founder
  - metal casters (73%)
  - metal smelting, converting and refining furnacemen (27%)

Two approaches to dealing with this variation by cleaning the data set were investigated:

- discarding all records containing descriptions with a significant level of multiple coding (511 records i.e. 0.8% of the data set)
- altering the non-primary codings, for those descriptions with multiple coding, to the most frequent coding

The rationale for the former is that such records should be prioritised for further human review for coding errors, while the rationale for the latter is that it might be assumed that the most frequent coding is correct and the others are errors.

Furthermore, some of the variation in coding may be legitimate; the examples in the table all appear to be possible valid interpretations. It is not known whether there was additional context available to the coders, or whether they made an arbitrary choice in such cases.

3.2 Edit Distance Classifier

The first automatic classifier tested was a relatively simple string similarity algorithm, the intuition being to select the HISCO class whose definition most closely matched the input string. The similarity measure used was *edit distance*: the edit distance between two strings is the number of single-character insertions, deletions or replacements needed to transform one into the other.

The simplest approach is to compare the description to be classified with all the HISCO definition strings, and to select the one with the smallest edit distance from the input. However, this would not work well for cases where the input string occurs as a substring embedded in the definition string, or vice versa. For example, the input string *machinery fitter* should probably be coded to the HISCO description *Machinery Fitters and Machine Assemblers*, which has a large edit distance from the former.

To address this issue, each of the words in the input string is compared pair-wise with each of the words in each candidate definition. The definition with the greatest number of close-matching words, as defined by edit distance, is selected.

3.3 Individual Machine Learning Classifiers

Using the Mahout machine learning framework (Apache Software Foundation, 2011), two individual machine learning classifiers were evaluated[1]. They were selected as the most mature and fully integrated of those supplied with Mahout:

- Stochastic Gradient Descent (SGD) (Zhang, 2004)
- Naive Bayes (NB) (Langley, Iba, and Thompson, 1992)

3.4 Ensemble Approaches

The ensemble approach (Dietterich, 2000) is based on the premise that better accuracy may result from combining classifications obtained independently from multiple algorithms. The following ensemble methods were evaluated, each combining results from some or all of the individual classifiers described in the preceding two sections:

- majority voting
- confidence-weighted (1)
- confidence-weighted (2)

The voting method used the three individual classifiers: edit distance, SGD and NB. The need for more than two participants to obtain a majority was one of the motivations for employing the string similarity classifier in addition to the machine learning classifiers.

The confidence-weighted ensembles combined the individual machine learning classifiers (SGD and NB). The general idea was to allow the overall decision of the ensemble to be influenced by a degree of confidence attached to each of the individual classifier decisions. However, while the SGD classifier generates an indication of confidence for each classification decision, the NB classifier does not, and so it was not possible to have an ensemble that simply selected the classifier decision with the highest confidence. Different approaches to addressing this problem were taken by the two confidence-weighted ensembles.

The first ensemble made a choice between the two individual machine learning classifiers based on the confidence expressed by SGD. If this was

[1] The Complementary Naive Bayes algorithm, which was also employed in the previous work on classification of causes of death, was dropped since it yielded very similar results to Naive Bayes in that work.

sufficiently high, the ensemble selected the SGD classification, and the NB otherwise. Thus the ensemble decision was driven by the confidence of the SGD classifier.

To do this, the relationship between the accuracy of the SGD classifier's decisions and its corresponding confidence levels was examined offline, by analysis of previous experimental results. A threshold value for the confidence measure was identified, above which the classifier's decisions were correct in 50% of cases.

During operation of the ensemble classifier, both SGD and NB classifiers were run on each input, and the SGD decision selected if its confidence value exceeded the fixed threshold, and the NB decision otherwise.

The approach taken by the second confidence-weighted ensemble was to synthesise an approximation to a confidence measure for the NB classifier, so that the ensemble could then select the decision of the individual classifier with the higher confidence.

To do this, the trained NB classifier was run offline over its own training set, to determine its relative accuracy on each output class. For each output class *X*, the proportion of records that it classified as *X* correctly was recorded.

During operation, the ensemble classifier took the decision of the NB classifier and looked up the corresponding correctness ratio. Interpreting this as a crude approximation to the probability that the NB classification decision was correct, the value was used as a confidence measure, allowing the ensemble to decide between the NB and SGD classifications.

3.5 Feature Selection

Due to the high dimensionality of the feature space in a text classification system, it is desirable to reduce the feature space without sacrificing classification accuracy. Terms that have no relationship to any given classification provide little information to the system and can often reduce the accuracy of the model as they introduce noise. Removal of low-value terms can be achieved using automatic feature selection, a process implemented here using the X^2 statistic (CHI) as described by (Yang and Pedersen, 1997) to calculate the 'goodness' of a term. Terms with a goodness value below a fixed threshold are removed from the feature set during the training process.

3.6 Swap Randomisation

Swap randomisation (Gionis et al., 2007) was used to calculate a baseline to act as a comparative value for the other results. Swap randomisation scrambles the training data by removing the relationships between the dependent and response variables, while leaving the distributional properties of the variables intact. It is this relationship between the dependent and response variables that the machine learning methods are trying to model. By removing this relationship and observing how this affects the results we can quantify how well the methods are actually performing.

If swap randomisation had little effect on the performance of our models then we would have to question whether our methods were doing anything useful. For example, if 90% of the dataset were classified to the same output class then our model could score highly by predicting only that class regardless of input, and would still score highly with swap randomised data. If however we noticed a marked decrease in performance after swap randomisation, then we could be more confident that our models were extracting meaningful patterns in the data.

3.7 Summary

To summarise, the approaches evaluated were:

- individual classifiers: string similarity, SGD, Naive Bayes;
- ensemble classifiers: majority voting, and two confidence-weighted versions;
- feature selection for the machine learning classifiers;
- data cleaning: correcting and discarding records that had been coded to multiple classes;
- evaluation on randomised data to give a baseline.

4 Evaluation

4.1 Sequence of Experiments

A first group of experiments measured the accuracy of the individual and ensemble classifiers, without feature selection or any cleaning of the data set. A second group assessed the effects of using feature selection and cleaning, with the same six classifiers. Finally, the accuracy of the best-performing classifier configuration from the

previous experiments was re-computed using several alternative accuracy measures.

4.2 Accuracy Measures

The measure used to report classification accuracy in most of the experiments is a straightforward exact match: the proportion of records in the test set for which the output of the classifier is the same as the class selected by the human coder.

Since the HISCO occupational coding has a hierarchical structure, it is also possible to consider other looser interpretations of correctness. HISCO has 9 'major' groups and 76 'minor' groups, a fragment of which is shown in Figure 1. As can be seen from the diagram, a worker may be correctly classified to various levels in the hierarchy. Thus a person may be correctly classified as either *62460 Horse Worker* or *624 Livestock Workers*.

The following alternative measures were also calculated for the best-performing classifier:

- match unit group
- match minor group
- match major group

These progressively relax the closeness of match required between the two classes for a classification decision to be considered as correct.

The *match unit group* measure considers a classification to be correct if it is in the same unit group as the class selected by the human coder. The other measures relax the condition further by allowing any matches within the same minor group or major group. The potential relevance of these measures is that for some purposes, researchers may only be interested in relatively coarse occupation classifications.

4.3 Validation of Results

There are several well-known ways to validate machine-learning results, including k-fold cross validation, random sub-sampling and the leave-one-out method (Kohavi, 1995; Witten and Eibe, 2005).

The reported results were calculated using the repeated random sub-sampling method. The data set was repeatedly partitioned in a random, non-sequential 80/20 training/testing split, resulting in a random sub-sample of the dataset being used for training and the complement of this random sub-sample being used in testing, for each repetition.

Unlike crossfold validation methods, the repeated random sub-sampling method has been shown to be asymptotically consistent, which can result in more pessimistic predictions of the test data compared with cross validation (Shao, 1993). However, this method has the obvious drawback that some data points may be included in the training set on every run, or never appear. This seems unlikely to be a problem given the size of our data set.

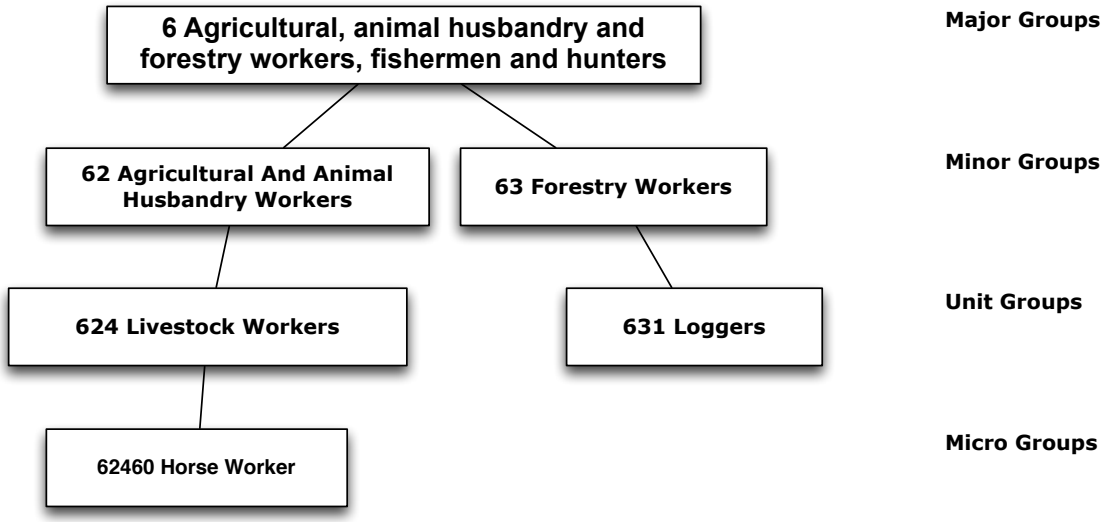

Figure 1. Example of HISCO hierarchy.

4.4 Results

Table 2 shows the exact-match classification accuracies obtained using the three individual classifiers over 5 independent runs, with 95% confidence interval. In each run, 80% of the records (51,200) in the HISCO-coded data set were randomly selected, and the remainder used for testing (12,800). For the machine learning classifiers, the 80% subsets were used for training. No feature selection or data cleaning was used.

Classifier	Accuracy
string similarity	30.8 ± 4.9%
SGD	22.1 ± 23.7%
Naive Bayes	90.1 ± 0.3%

Table 2. Classification accuracy of individual classifiers.

As expected, string similarity matching performed significantly better than chance, but still not particularly well. SGD performed very inconsistently, with accuracy ranging between 4% and 50% in individual runs. Naive Bayes performed consistently well, and indeed the results here were only improved upon slightly in subsequent more complex approaches.

Table 3 shows the exact-match classification accuracies obtained using the three ensemble classifiers, using the same procedure as in the previous set of experiments.

Classifier	Accuracy
majority voting	90.1 ± 0.3%
confidence-weighted (1)	90.2 ± 0.3%
confidence-weighted (2)	38.9 ± 4.2%

Table 3. Classification accuracy of ensemble classifiers.

None of the ensembles were able to improve on the performance of Naive Bayes. Rather than being able to combine the strengths of individual classifiers, the ensembles were dominated by Naive Bayes, which performed so much better than the others.

The voting ensemble produced the same results as Naive Bayes, due to an accident of implementation whereby that classifier is picked whenever there is no agreement. The first confidence-weighted ensemble also yielded the same results; clearly SGD was never sufficiently confident to outweigh Naive Bayes. The second confidence-weighted ensemble performed much worse, precisely because the SGD decision was sometimes selected.

Table 4 shows the effect of enabling feature selection for the relevant individual and ensemble classifiers.

Classifier	Accuracy
SGD	36.0 ± 12.1%
Naive Bayes	91.8 ± 0.3%
majority voting	91.8 ± 0.3%
confidence-weighted (1)	91.7 ± 0.2%
confidence-weighted (2)	34.0 ± 5.4%

Table 4. Classification accuracy with feature selection.

This yielded a significant improvement to SGD, and a small but useful improvement of around 1.5% in Naive Bayes.

Table 5 shows the effect of performing initial data cleaning by correction for the individual and ensemble classifiers, without feature selection. The classifications of 364 records in the data set were altered from an alternative classification to the most popular one for that occupation description. This represents 0.6% of the data set.

Classifier	Accuracy
string similarity	25.6 ± 2.0%
SGD	25.4 ± 1.1%
Naive Bayes	90.8 ± 0.3%
majority voting	90.8 ± 0.4%
conf-weighted (1)	90.8 ± 0.4%
conf-weighted (2)	31.9 ± 1.9%

Table 5. Classification accuracy with data cleaning by correction.

Correction of multiply-coded descriptions yielded a slight improvement in performance of SGD, and a large improvement in consistency. For Naive Bayes it gave a very marginal improvement.

Table 6 shows the effect of performing initial data cleaning by correction and by discarding, for the individual and ensemble classifiers, with feature selection enabled. Records were discarded from the data set for those descriptions that occurred more than 10 times and were coded to alternative classifications in more than 10% of cases. 511 records were discarded, representing 0.8% of the data set.

Classifier	Accuracy	
	Corrected	Discarded
SGD	23.3 ± 11.4%	19.6 ± 25.1
Naive Bayes	92.4 ± 0.3%	92.3 ± 0.2
majority voting	92.4 ± 0.3%	92.2 ± 0.3
conf-weighted (1)	92.4 ± 0.4%	92.3 ± 0.3
conf-weighted (2)	27.7 ± 3.7%	31.7 ± 3.7

Table 6. Classification accuracy with feature selection and data cleaning.

In this case feature selection together with correction reduced SGD performance, whereas a further slight improvement was obtained for Naive Bayes. Discarding records gave worse results for SGD and the same for Naive Bayes.

Table 7 shows the effect of using alternative accuracy measures to calculate the performance of the best-performing classifier from the previous experiments (Naive Bayes with feature selection and cleaning by discarding). As expected, each successive relaxation of the correctness condition yielded a small increase in accuracy.

Measure	Accuracy
exact match	92.3 ± 0.2
match unit group	92.2 ± 1.2
match minor group	93.3 ± 1.4
match major group	94.9 ± 1.4

Table 7. Classification accuracy using alternative accuracy measures.

Table 8 shows the effect of using swap randomisation on the training data to remove any relationship between features and classes. The classifiers all perform extremely poorly on such data, which indicates that the results observed for the real data are indeed based on modeling genuine patterns.

Classifier	Accuracy
SGD	0.6 ± 0.3%
Naive Bayes	2.5 ± 0.2%
majority voting	2.5 ± 0.2%
confidence-weighted (1)	2.6 ± 0.3%
confidence-weighted (2)	13.2 ± 0.6%

Table 8. Classification accuracy using randomized dataset.

5 Conclusions

The best automatic classification accuracy was achieved using the individual Naive Bayes classifier with feature selection enabled, and cleaning of the data by correction of multiply-coded descriptions. This yielded exact-match accuracy of 92.3 ± 0.2%, ranging up to 94.9 ± 1.4% when considering only major group matching. Currently this appears to be the most promising approach for occupation coding of the full Scottish data set.

The performance of the ensemble approaches was disappointing, yielding no improvement over the individual Naive Bayes classifier. It is possible, though, that there is still benefit to be had from an ensemble approach if other individual classifiers with performance comparable to Naive Bayes can be found. Although the SGD classifier performed very poorly here, as shown in Table 2, it did perform well on some data sets in our previous work on cause of death classification. This leads us to think that further investigation of its pathological behaviour here would be worthwhile.

We plan to develop this work in a number of other ways:

- Perform a manual review of the incorrectly classified records, looking for any common patterns that might be candidates for additional cleaning steps.
- Experiment with other data cleaning measures such as spelling correction.
- Experiment with other string similarity metrics, and add the ability for such classifiers to generate proxy confidence values, allowing them to be incorporated into confidence-weighted ensembles.
- Repeat our experiments on other data sets, in particular the full Cambridge set coded to SOCH.
- Investigate the applicability of this work to the classification of text fragments from other domains of genealogical interest, such as family names (the problem in the latter case being to code to standard spellings).

Regarding the methodology for performing classification of a given large-scale data set, the scope for automating the following will be investigated:

- The selection of the most suitable classifier (whether individual or ensemble, and with or without spelling correction using various dictionaries), guided by experiments on samples from the data set.

- The determination of the minimum size training set then required to be hand-coded, to achieve a given acceptable level of accuracy.

In conclusion, the use of machine learning classifiers and ensembles appears to be a potentially promising method for coding large data sets. We are continuing to investigate how to raise accuracy, and how to automate the overall process of selecting a classifier, deciding training set size, running the classification and validating the results.

Acknowledgements

This work was supported by ESRC grant ES/K00574X/1 Digitising Scotland. We thank the anonymous referee for their very helpful comments.

References

Apache Software Foundation. 2011. Apache Mahout: Scalable Machine Learning and Data Mining. *http://mahout.apache.org/*.

Wendy Bottero and Kenneth Prandy. 2001. Women's Occupations and the Social Order in Nineteenth Century Britain. *Sociological Research Online*, *6*(2).

Bureau of Labor Statistics. 2010. Standard Occupational Classification (SOC) System. US Bureau of Labor Statistics. *http://www.bls.gov/soc/*.

Jamie Carson, Graham Kirby, Alan Dearle, Lee Williamson, Eilidh Garrett, Alice Reid, and Christopher Dibben. 2013. Exploiting Historical Registers: Automatic Methods for Coding C19th and C20th Cause of Death Descriptions to Standard Classifications. *Proceedings New Techniques and Technologies for Statistics 2013*, 598–607.

Thomas G Dietterich. 2000. Ensemble Methods in Machine Learning. *Multiple Classifier Systems*, 1857:1–15. Springer.

Aristides Gionis, Heikki Mannila, Taneli Mielikäinen, and Panayiotis Tsaparas. 2007. *Assessing Data Mining Results via Swap Randomization*. *Transactions on Knowledge Discovery From Data (TKDD*, *1*(3) (December).

HISCO. 2013. HISCO Tree of Occupational Groups. *http://historyofwork.iisg.nl/major.php*.

Ron Kohavi. 1995. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *Proceedings 14th International Joint Conference on Artificial Intelligence*, 1137–1143.Morgan Kaufmann Publishers Inc.

Pat Langley, Wayne Iba, and Kevin Thompson. 1992. *An Analysis of Bayesian Classifiers*. Proceedings AAAI-92, 223–228.

Marco H D van Leeuwen, Ineke Maas, and Andrew Miles. 2002. *HISCO: Historical International Standard Classification of Occupations*. Leuven University Press.

Ken Prandy and Paul Lambert. 2012. CAMSIS: Bibliographic Review. *http://www.camsis.stir.ac.uk/review.html.*

Jun Shao. 1993. Linear Model Selection by Cross-Validation. *Journal of the American Statistical Association*, 88(422), 486–494.

Ian Witten and Frank Eibe. 2005. Data Mining: Practical Machine Learning Tools and Techniques. 2nd ed. Morgan Kaufmann.

Yiming Yang and Jan O Pedersen. 1997. A Comparative Study on Feature Selection in Text Categorization. *Proceedings 14th International Conference on Machine Learning*, 412–420. ACM.

Tong Zhang. 2004. Solving Large Scale Linear Prediction Problems Using Stochastic Gradient Descent Algorithms. *Proceedings 21st International Conference on Machine Learning*, 116–123. ACM.