From Manual to Machine: challenges in machine learning for COICOP coding

Susie Jentoft, Statistics Norway
Boriska Toth, Statistics Norway
Daniel Müller, Norwegian University of Life Sciences

Abstract

The classification of data is a time-consuming task performed by most statistical bureaus. Manually converting text to classifications can provide good quality data but requires both expert coders with good knowledge of the standards and many resource hours. These problems are amplified as larger data sources are incorporated into official statistics. Advancements in machine learning algorithms and increased accessibility to these are allowing new opportunities for classification workflows.

Here, we provide a case study for using machine learning algorithms to classify COICOP (classification of individual consumption according to purpose) in the Norwegian Household Budget Survey (HBS). The 2022 survey represents a new paradigm for the Norwegian HBS in combining a sample survey with novel big data sources and underscores the need to automate the classification process in modern surveys. Machine learning can greatly reduce the burden of manual labelling for the HBS, but it does not eliminate the need for human labellers to classify and quality check items newly appearing on the market. This challenge of deploying an automated system on evolving data is common to many machine learning problems in the production of official statistics. We define a human-in-the-loop paradigm for implementing machine learning in conjunction with human labelling, in which the two processes support each other synergistically. We evaluate the performance of machine learning algorithms by reporting how the savings in manual labelling trades off with accuracy.

Different algorithms were tested including random forest, logistic regression, support vector machines (SVM) and XGBoost. Overall, SVM performed the best for predicting COICOP classification for transaction data. Different sets of features including goods name, group names, ingredients (food only) and price were tested for their prediction performance. Including the additional group names and goods price increased the prediction performance (from an accuracy of 83% to 90% on a hold-out test set for classifying foods) and should be considered for implementation. The performance of COICOP classification for the noisier data of user-scanned receipts with item names identified by optical character recognition, a more modest performance of around 60% accuracy was found when classifying both foods and non-foods using only the good's name.

A major hurdle within supervised machine learning is access to good quality training data. Data from heterogenous auxiliary sources at Statistics Norway were used to generate a large dataset of item names with COICOP labels. However, future work is need to improve the balance of different COICOP classes in the training data, better incoporate training data selectively from the very large but noisy auxiliary sources, and supplement machine learning with rule-based methods.

Keywords: Supervised machine learning, classification, household budget survey, language processing.

1. Introduction

Classifying text strings into official classifications is a common task performed by the official statistics community. Text data may be collected directly, in the case of survey data, and entered by interviewers or the respondents themselves. Increasingly, in the current age of multiple data source integration, secondary data may also require classification. In this case, data is often on a larger scale and less organised than that collected by a traditional interview survey.

With the increased computing power over the last decade, machine learning techniques have become more accessible. The Blue skies thinking network (2018) concluded that Machine learning techniques can and should be used, in certain cases, for processing of both secondary and primary (survey) data. Classification problems are a prime example of when machine learning techniques can help automate production processes (UNECE, 2021). Not only can they aid in the modernisation of the statistical production process but allow the use of big data sources, which under the conditions of manual coding, would essentially not be feasible.

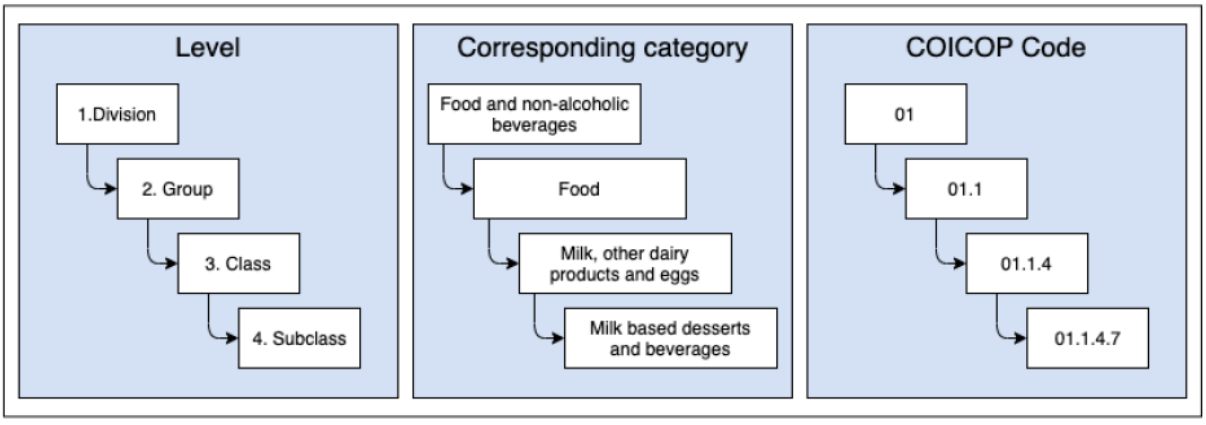

Figure 1. Example of hierarchical structure of COICOP classification. Source: Müller, 2021.

The Household Budget Survey (HBS) is a European-wide survey which provides a detailed picture of households' annual consumption (Eurostat, 2022). The survey provides important macroeconomic indicators and was last run in Norway in 2012 (Holmøy & Lillegård, 2014). Goods and services are classified into groups using a classification known as COICOP (Classification of Individual Consumption According

to Purpose). This is a hierarchical classification system with up to 5-digits of detail (Figure 1) (UN, 2018). In the previous Norwegian HBS items that respondents in the sample consumed were entered manually along with their COICOP classification.

In 2022, this survey represents a new paradigm for the Norwegian HBS in combining a sample survey with novel big data sources and underscores the need to automate the coding process in modern surveys. Respondents in the modernized survey can choose to scan in receipts or manually enter items via a smartphone app. Furthermore, automating COICOP coding enables the incorporation of big data sources from private enterprise, including detailed purchase transaction information for all purchases made by debit card in the year of the survey from all of Norway's major supermarket chains and retailer stores. These purchases can be linked to the demographic group of the purchaser using a privacy preserving, pseudonymized process of matching store records of transactions to bank records of debit card transactions occurring at the same store with the same timestamp. The use of these massive new data sources in an automatized workflow can allow for far more frequent publishing of HBS statistics as well as more fine-grained information on purchasing habits in specific demographic groups, regions, and times of year, due to the vast amount of data in each stratum.

There are two sources of data in the 2022 HBS for which there is a need to automate COICOP classification of purchased items: purchase transaction records generated at store cash registers, and receipts scanned by respondents in the survey.

Purchase transaction records

Statistics Norway is obtaining records of every item purchased at the three largest grocery chains in Norway in the whole calendar year of 2022, recorded at cash registers. We expect a dataset of roughly 300,000 unique items. These transaction records contain detailed information of the sort that appears on receipts, including the name of the item, store, date and timestamp of the purchase, tax, link to other items on the same receipt, etc.

Scanned receipt data from the survey

Respondents chosen for HBS 2022 were asked to submit information about every item they purchased during a one-week period. They could choose to manually enter

information into a phone app or scan in a receipt. The analysis and results in this paper refer to data from HBS 2022 that has arrived between January and June 2022. About 90% of items were entered by respondents via scanning a receipt. The receipts were transcribed into text using optical character recognition, and in many cases, there are errors in the process. The resulting dataset includes the item name, information about the respondent, store, and date of purchase. There are roughly 33000 unique items out of 150,000 total items that have arrived by midyear of the survey.

All data sources for the HBS contain text strings of the name of the goods and services that require classification into COICOP. This provides an excellent example of opportunities for machine learning in classification problems and is the focus of this study. The use of machine learning for the classification of text strings to COICOP in the HBS, however, has several challenges. Three of these we discuss in this study:

- Obtaining “gold standard” training data and integrating lower quality training data from auxiliary sources
- Developing the model
- Deciding the appropriate level of automation vs manual coding

These challenges are not unique for our example using HBS data but are common for these types of problems.

Human-in-the-loop (HIL) workflow, and ways to evaluate HIL models for automating classification.

The classic machine learning paradigm involves a two-step process of: training a model on training data, and testing the model on test data that is presumed to be unseen at the time of training but drawn from the same distribution as the training data. Train and test sets for model learning and evaluation, respectively, are typically generated in practice by randomly splitting a labelled set into training and hold-out sets. For the problem of automating a coding process faced by NSI’s, this classic machine learning paradigm does not capture either how the automation process is implemented nor the measures of performance that are relevant in practice. While the classic paradigm views data as coming from a static process, data generating processes evolve with time in practice. For this reason, HBS 2022 considers it essential for the

foreseeable future that there is a “human in the loop” doing some manual checking of codes as new items arrive or categories change with time in ways a model can’t anticipate, as opposed to implementing a fully automated classification process. When it comes to automating a coding process at an NSI, one can think of the human-in-the-loop paradigm for developing a model and evaluating its performance in relevant ways as the following three steps:

1. Develop a model using training data from a specified time period and define a target source of data from a specified time period that needs to be coded and from which the test set will be drawn at random. This target source often involves data generated later than the data in the training sets. For HBS 2022, for instance, the target sources to be coded are items from transactions and survey receipts for purchases made in 2022.
2. Define a human-in-the-loop procedure for how items in the target data source will be coded, where in practice some items will be checked by a human labeller while much of coding is automated. One example of a procedure might be that for previously unseen items (i.e. those not closely matching items in the training data), those with a prediction probability (a confidence measure returned by the machine learning algorithm) above some threshold, T, are predicted by the algorithm, and those below the threshold labelled manually. For items in the target data source that appear in the training set, a procedure might be to use the predicted code if it matches the code in the training set, and to check manually otherwise.
3. Test the human-in-the-loop procedure, reporting both a performance measure for the automated coding (i.e. accuracy or F1) and for the burden of human labelling (number of items that needed to be checked manually, or an estimate of time needed for manual labelling).

Such a procedure of training a model, then generating predictions on new data and choosing some items to check manually becomes an iterative process as the model is retrained on the manually checked labels. This is shown in Figure 2.

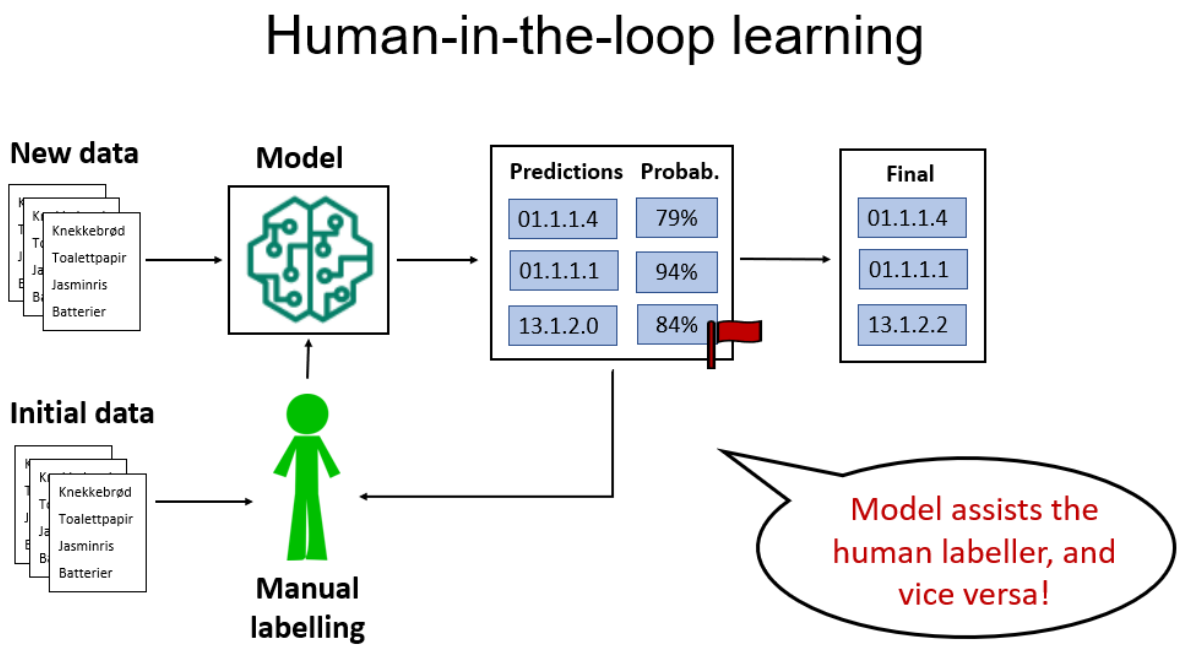

Figure 2. Diagram of the workflow behind a human-in-the-loop system.

This process of evaluating the performance of the human-in-the-loop model on an evolving newly arriving data source captures the fact that what we're really interested in is how much of the burden of manual labelling machine learning can remove, at what trade off with accuracy.

Machine learning can support the human manual labeller by identifying the items the model was least confident on and for which manual labelling has high impact for improving the model. In addition, we found that simply marking and sorting items by predicted COICOP group could significantly speed up the manual labelling process.

2. Training data

There are five main sources of training data: CPI transaction records, supermarket transaction records, survey receipts collected during the survey, dictionaries linking COICOP groups to keywords, and imports data. For the problem of COICOP prediction on transaction data, the labelled transaction data is a rich enough source that it is sufficient as a training set. For the problem of predicting on scanned receipts data, the

relatively small amount of labelled receipts data must be supplemented with other, heterogenous sources.

2.1. *Background on the training data*

CPI transaction data

The division for prices at SSB uses transaction data from supermarkets to estimate the CPI for some of their partial indexes. The process of classifying these goods by COICOP has previously been done manually; however, new items are now classified based on a combination of machine learning prediction algorithm (for food) and keyword/rule based for non-food items. Items predicted with low probability are checked and edited manually each month. The data in this study is based on a catalogue of items sold up until January 2021 and contains around 30 000 food and non-food items. Nearly 24 000 additional non-food items were provided by the CPI part way through this study and are included in the training dataset for survey data.

While these goods have COICOP classifications, they are provided in the European ECOICOP standard. These were converted to the new 2018 UN COICOP standard for use in HBS. 229 of the 303 ECOICOP groups represented in the file could be mapped to a single UN COICOP group. The remaining groups were converted using a more detailed national 6-digit COICOP and various other variable such as keywords in the ingredient list.

Supermarket transaction data

Details on around 300 000 unique products sold in 2018 were obtained from Norwegian supermarkets for use in testing and development. While these were non-labelled data, some were able to be linked to additional sources giving an alternative group coding (GPC and ENVA). In some cases, these groups provided a direct link to a COICOP classification. A number of items were also manually classified to build on the training dataset. This includes 500 items originally manually coded directly to COICOP classification based on high prices or sales volume, or low probability from the prediction models. An addition 2500 food items were manually coded/recoded by a group from the University of Oslo looking at nutritional information on Norwegian food purchases in a parallel project. All manual coding was done directly into the 2018 UN COICOP classification.

Survey receipts data

There are two sources of training data that involve receipts obtained from respondents in the survey: data from so-called manually coded receipts where respondents chose to manually enter receipt information into a web app, including the item name and 5-digit COICOP category; and an additional 3000 manually labelled items from scanned receipts from the ongoing HBS 2022 survey.

Dictionaries of keywords

We included several dictionaries that link COICOP codes (in the new classification) to Norwegian keywords that describe the group. One of these of size 2500 is a Norwegian translation of a reference dictionary distributed by the United Nations Statistical Division that uses English keywords to describe and give examples for COICOP groups. Another is a list of 2400 keywords linked to 500 COICOP codes, which serves as a dictionary of search terms that give survey respondents suggested COICOP codes when they type keywords.

Auxiliary imports data

We had access to a very large auxiliary dataset that could be used to link goods to COICOP groups from the foreign trade section at Statistics Norway. This dataset corresponds to the TVINN system of customs declarations for all items imported to Norway, and item names could be linked to COICOP codes using a 3-step conversion process and retaining the 1.5 million items which mapped to a unique COICOP code.

2.2. *Unbalanced nature of the training data*

Figure 3 shows the items counts in some of the components of the training data. It reflects several challenges: the imbalanced representation of COICOP groups in the different sources for the training data, and the sparsity of some groups.

Figure 3. Item counts by source. Top left: Survey receipts source; Top right: CPI transaction data (including predicted). Bottom left: Additional non-food CPI transaction data. Source: Müller, 2021.

3. Model building

Building a machine learning pipeline involves multiple choices. Models can be improved by changes in the pre-processing of the text, representation of the text and weighting, choice of features, the algorithm used and tuning of the parameters. Here we discuss some of the many choices we made and give results from a comparison of different algorithms and feature selections. Cross- validation was used for parameter fitting.

3.1. Pre-processing

The main text variable used for classification is the good's name. This is available in all the data sources; however there are some differences. For example, the goods' names from the survey receipt data are generally shorter with more abbreviations compared to the other sources. All good names were pre-processed with the following general steps used:

- Replaced weights and volumes with 'solid' and 'liquid'
- Removed single letter words and numbers (can be a problem in non-food – eg shoe size)
- Removed brand names (for transaction food prediction)

- Tidy -convert all to lower case, remove double spaces etc.

3.2. Representation

Algorithms can't process text strings directly. The processed text variables need to be vectorized into a matrix for using in classification models. The most common form for this, which is used in this study is using 'bag-of-words'. The Bag-of-Words is a simple way to represent text where each word is represented as one dimension of a numerical feature vector (Figure 4). Sometimes, due to differences in spelling etc. it is best to use parts of a word rather than the full word. This is referred to as n-grams. N-grams are sequences of n consecutive units in a text, typically sequences of words or characters. We found 2- and 3-grams of characters to work best in the vectorising.

Key	Word
0	coop
1	jasminris
2	toalettpapir
3	økonomi
4	lambi
5	extra
6	long
7	husman
8	knekkebrød

Item descriptions
i_1 = coop jasminris
i_2 = toalettpapir økonomi
i_3 = lambi toalettpapir extra long
i_4 = husman knekkebrød økonomi

Feature vectors
X_{i1} = [0.71, 0.71, 0, 0, 0, 0, 0, 0, 0.]
X_{i2} = [0, 0, 0.71, 0.71, 0, 0, 0, 0, 0.]
X_{i3} = [0, 0, 0.41, 0, 0.53, 0.53, 0.53, 0, 0.]
X_{i4} = [0, 0, 0, 0.49, 0, 0, 0, 0.62, 0.62]

Figure 4. Example of item description and bag-of-words vector. Source: Müller, 2021.

Finally, some n-grams will be more important than others. We used Term Frequency - Inverse Document Frequency ($tfidf$) to weight the n-grams in some experiments. This is commonly calculated by comparing the number of occurrences of a word/n-gram in a single good's name to its usage in the data set. A common way to calculate $tfidf$ is given in Raschka and Mirjalili, (2019, p. 265) as

$$tfidf(d,t) = tf(d,t) \cdot idf(d,t)$$

where tf is the term frequency for term t (word/n-gram) in document d (good's name) and

$$idf(d,t) = log\left(\frac{nd}{1+df(d,t)}\right)$$

where n_d is the number of documents (good's names) in the data set and $df(d,t)$ is the number of documents, d, that contain the term, t.

3.3. Algorithms

Using machine learning algorithms to classify goods to COICOP has been in production at Statistics Norway since 2017 (Myklatun, 2019). It has been utilized for some supermarket transaction data to establish a dictionary of goods and their COICOP groups for the consumer price index (CPI). Threshold values are used to pass some goods automatically into the dictionary while those below the threshold are coded manually. In earlier studies, support vector machines (SVM) provided the best prediction power over random forest and logistic regression (Myklatum, 2019). Algorithms including XGBoost, SVM, Random forest and Logistic regression were tested in Python using the *scikit-learn* and *xgboost* modules.

3.4. Performance metrics

Three metrics were used to assess the performance. Accuracy was defined as

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

where TP is the number of true positives (positive cases that were predicted correctly), TN is the number of true negatives, FP is the number of false positives and FN is the number of false negatives.

F_1-score is a popular metric for comparing models, and it is especially useful for assessing performance on unbalanced data. Precision and recall are defined as

$$Precision = \frac{TP}{TP+FP} \qquad Recall = \frac{TP}{TP+FN}$$

The F_1score is then a combination of these

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision+Recall}$$

This is calculated for each COICOP category and averaged either as a weighted average or a macro-average.

3.5. *Benchmark for accuracy of classification*

From most machine learning algorithms, it is possible to obtain a probability or certainty measure. These can be used as a threshold in a human-in-the loop type implementation, to trade off the accuracy of the automatically labelled data with the burden of manual labelling. Deciding the threshold to use should be considered carefully and will depend on resources available and the quality of the data and model.

An important consideration is that two expert manual coders will likely code the same good to different classification in some cases. This happens both when text descriptions are inadequate and when classification systems are not defined precisely enough. Over time, both concepts and data may change, and classification frameworks don't always reflect this accurately and allow for one aligned, perfect "gold standard". An earlier double-coding study used a small dataset of around 100 goods with two coders independently classifying items according to the 2018 COICOP. The study showed a consensus rate of around 87 percent (Jentoft, 2021 unpub.). This indicates that a goal for a machine learning algorithm achieving 100% accuracy is likely unachievable and an unnecessary aim. If a ML model can achieve a similar accuracy of close to 90% found in the double-coding study (evaluating the model against a human coder's labels), this would indicate precision on par with a human coder.

4. Results for predicting COICOP for supermarket transaction data

4.1. Model algorithm

Here we test four common algorithms (Support Vector Machine, Random Forest, Logistic regression and XGBoost) using training data for food based on CPI transaction and supermarket transaction (manually and group labelled) data. The training data here included only these two sources as they had the closest alignment with that for predicting the unlabelled supermarket transaction data. The algorithms were tested first using only the goods name, converted using 3-gram characters. A 20 percent hold-out dataset (same for all) was used to compare the algorithms.

For Random Forest, 1000 trees were learned, and trees were grown until nodes were pure. A standard feature sample size was the square root of the number of features. For XGBoost only 50 estimators were used with a maximum depth of 4 (due to the long running time).

Table 1 gives the accuracy and F_1-scores for the hold-out test set. It shows that the Support Vector Machine (SVM) was by far the fastest running.

Table 1 Comparison of four algorithms for predicting COICOP using 3-gram good's name

	Run time (seconds)	Accuracy	F1-macro	F1-weighted
SVM	3	0.83	0.79	0.82
RF (1000 trees)	1094	0.76	0.70	0.76
Logistic regression	446	0.77	0.67	0.76
XGBoost	4146	0.73	0.69	0.73

4.2. Feature selection for food classification

In several previous studies, the name of the goods were the only features in prediction models. The CPI additionally uses grouping codes (ENVA) which are unique across the supermarket chains. Here we explore additional features available for some of the training data including:

- Group names (text string) for ENVA groups
- Group names (text string) for GPC groups
- Ingredients list (text string)
- Average price (numeric)

Different combinations of these were tested using SVM and a 20% hold-out set and are shown in Table 2 . A subset of 5000 features were selected (or 5001 when including price). These were chosen based on the term frequency of the n-grams in the dataset using the function *CountVectorizer* from the scikit-learn module in Python (Pedregosa et. al., 2011). A subset of the training data was used for feature selection where only data that had a price specified was included (around 2400 items were thus excluded).

Table 2. Comparison of model metrics from a 20 percent hold-out set using different features.

	Accuracy	F1-macro	F1-weighted
Goods name	0.83	0.79	0.82
Goods name + GPC + ENVA	0.89	0.86	0.89
Goods name + ingredients	0.84	0.80	0.84
Goods name + GPC + ENVA + ingredients	0.89	0.86	0.89
Goods name + GPC + ENVA + price	0.90	0.84	0.90

Overall, we saw a good improvement in the algorithm when using the name of the GPC and ENVA food groupings (Table 2). Including ingredients did not appear to improve the model prediction. Including price as a feature appeared to improve the model slightly and had the highest accuracy and weighted F_1-score.

4.3. Using probabilities for COICOP prediction of food in transaction data

Here we use a calibrated cross-validation measure from the best SVM (using as features good's name, ENVA, GPC, price) to estimate the probability for the COICOP classes. We used Platt's method which is a way of transforming the outputs from the SVM classifier into probability distribution over the classes. The method was implemented using the function *CalibratedClassifierCV* from the *scikit-learn* module.

For the supermarket transaction data, there are over 300 000 unique goods for 2018, of which around 116 000 are food items. Of these, around 24 000 were able to be directly linked to an item in the trainings data set and classified directly. The remaining 92 000 foods were missing COICOP labels and the SVM model was used to label them.

In Figure 5 we show the accuracy of the predictions at different threshold values for the model probabilities in the hold-out training dataset. The accuracy represents that for all observations with *at least* the specified probability level. If we choose an accuracy of 90 percent to find a corresponding probability threshold value, all predicted COICOP classifications with probabilities over 0.2417 can be used. In the training data set, using this threshold equates to 99 percent of the data. Using this same threshold on unlabelled data for food transactions, we see that the distribution of probabilities is not as high as that for the training data but is still quite high, with 94 percent of the data above that threshold (Figure 6).

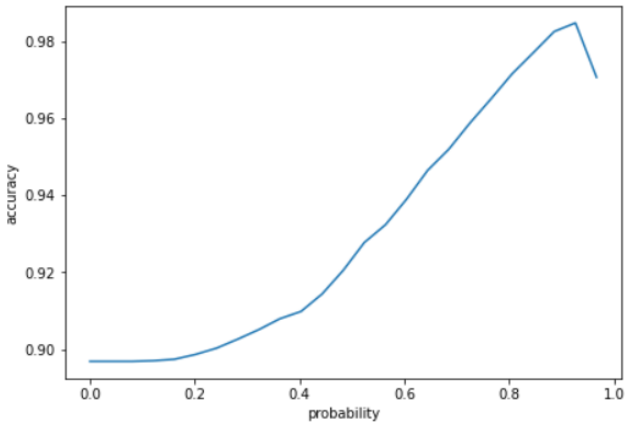

Figure 5. Accuracy at different threshold values for the prediction probability of the hold-out training data set.

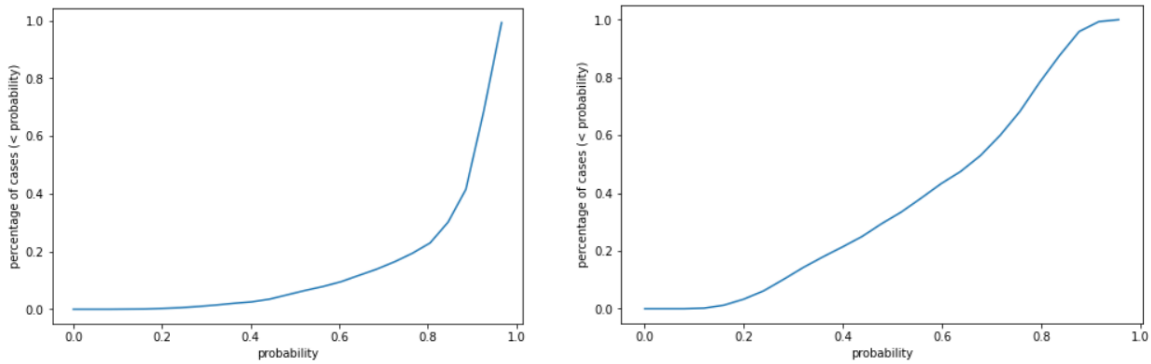

Figure 6. Percentage of cases with probabilities at least that value plotted against the various threshold values for probability. Left: training data. Right: unlabelled data

5. Survey data

To test the performance of machine learning on COICOP classification of scanned receipts from the survey, we constructed a test set of 1000 randomly sampled items scanned by respondents in a pilot survey conducted in June 2021. This set was manually labelled at SSB.

All sources of data listed in section 2.1 were initially used to construct a training set. The inclusion of the1.5 million item TVINN customs imports dataset was not found to improve performance overall, so that was left out when running the experiments below due to its size. In the results shown here, only the pre-processed item name was used to generate features. Classifiers were trained using logistic regression and random forest. For logistic regression, 3-character grams with unit weights (not tf-idf) were found to perform best, while for random forest, it was 2- and 3-character grams and

unit weights. The model parameters were tuned using cross validation. Logistic regression was found to perform just slightly better than random forest, at 59% accuracy for the former. This modest performance is expected to improve once the suggestions for future work described below are incorporated.

Table 2. Performance on a randomly chosen test set of scanned receipts from the pilot survey.

	Extractor	F1-weighted	Accuracy
Logistic regression	good's name, CV-ch33	0.585	0.592
Random forest	good's name, CV-ch23	0.573	0.583

We also tested the performance on a human-in-the-loop implementation. Choosing a subset of the test set having the prediction probabilities above a threshold that corresponded to 90% accuracy in predicting COICOP, only 18% of items could be chosen.

Percentage of the data automatically labelled at given thresholds for accuracy.

	Threshold	Percent above	Accuracy
Random forest, good's name, CV-ch23	.64	**42%**	**0.80**
	.92	**18%**	**0.90**

6. Conclusions

This study highlights some of the challenges faced while trying to build text-based classification models. When using relatively homogenous training data based primarily on previously classified CPI data, we were able to achieve prediction models with high accuracy in hold-out testing, on par with inter-coder agreement in the double-coding experiment. However, when these models were applied to unlabelled supermarket transaction data, prediction probabilities indicated that there were distributional differences between these groups. The distribution of the prediction probabilities was lower for the unlabelled data and around 6 percent had prediction probabilities lower

than the 90 percent accuracy threshold. While this is a small part of the data, future work includes building up the training data and manual coding is still warranted in a human-in-the-loop system.

Classifying the survey data to COICOP has its own set of challenges. The nature of the text in the good's name is somewhat shorter and therefore isn't as well harmonized with the CPI transactions training data. Additionally, we looked at classifying both food and non-food items, increasing the number of classification groups significantly. We therefore integrated additional sources into the training data to broaden the text features and increase the number of observations in each of the COICOP groups. One major challenge when training models for the survey data was how to incorporate these heterogenous sources of differing quality. The use of the 1.5 million item TVINN dataset provided no real improvement in performance overall. However, it would be interesting to see if this large but noisy source can nevertheless yield gains by selecting a targeted subset to include in the training set. For instance, one might include items with COICOP codes that have low representation in the rest of the training set, or items that score highly on a matching measure to at least one item in a dataset of transaction or survey items. The choice of weighting of the different components of the training data was also found to have an impact on performance. Weighting schemes can give larger weight to more reliable sources, or they can match the distribution of COICOP codes expected in the test set. Using store as a feature, possibly crossed with important textual features from the n-gram vectorization, has lots of potential. Finally, it is worth experimenting with supplementing machine learning with rule-based methods. For example, items sold at bookstores that presumably refer to titles should be coded as books and not using text-based prediction.

References

Eurostat (2022) *Household Budget Surveys (HBS) – Overview.* https://ec.europa.eu/eurostat/web/household-budget-surveys (accessed: 24.06.2022)

Holmøy & Lillegård (2014) *Forbruksundersøkelsen 2012. Dokumentasjon av datainnsamling, analyse av datakvalitet og beregning av frafallsvekter.* Statistisk Sentralbyrå

Jentoft, S. (2021 unpub.). *Double coding for COICOP of non-food items in transactions data.*

Myklatun, K. H. (2019) *Utilizing Machine Learning in the Consumer Price Index*. NSM 2019 Conference paper

Müller, D.M. (2021). *Classification of Consumer Goods into 5-digit COICOP 2018 Codes.* Master's thesis, Norwegian University of Life Sciences, Faculty of Chemistry, Biotechnology, Food Science.

Pedregosa *et al.*, (2011). Scikit-learn: Machine Learning in Python, *JMLR* 12, pp. 2825-2830

Raschka, S., & Mirjalili, V. (2019). Python machine learning: Machine learning and deep learning with python, scikit-learn, and tensorflow (3rd). Packt Publishing Ltd.

UNECE; Modernstats Blue skies thinking, (2018) Blue skies thinking*; The use of machine learning in official statistics*

UNECE (2021) Machine learning for Official Statistics