

Proceedings of Statistics Canada Symposium 2024: The Future of Official Statistics

Life in the FastText Lane: Harnessing Linear Programming Constrained Machine Learning for Classifications Revision

by Justin Evans and Laura Wile

Release date: September 8, 2025



Statistics
Canada

Statistique
Canada

Canada

Life in the fastText Lane: Harnessing Linear Programming Constrained Machine Learning for Classifications Revision

Justin Evans and Laura Wile¹

Abstract

Statistics Canada's Labour Force Survey (LFS) plays an essential role in the estimation of labour market conditions in Canada. Periodically, LFS revises its data to the most recent industry and occupational classification versions. Differences in versions can be extensive, including high-level and unit-group structural changes, creations, deletions, split-offs and combination of classification units (classes). Historically, to reconcile split-off classes - where one class splits into multiple classes - a sample of LFS split-off records would be manually recoded to the new classification version. Based on the split-off proportion observed in the recoded sample, a random allocation method would be applied on all data to reflect the changing Canadian labour market over time. This article proposes using machine learning (fastText), constrained to split-off proportions using linear programming, to revise industry and occupation classifications in LFS. The hybrid framework benefits from a text-based revision mechanism while adhering to traditional proportions driven estimates, thus ensuring a minimal impact on the comparability of published labour market indicators.

Key Words: Machine learning; Labour Force Survey; Classification revision; fastText.

1. Introduction

Statistics Canada's Labour Force Survey (LFS) plays a fundamental role in the estimate of labour market conditions in Canada. Periodically, LFS data is revised to reflect the most recent classification version, seasonal adjustment, or to introduce methodological enhancements. Revisions ensure that survey estimates continue to reflect the size and composition of the Canadian labour market, while minimizing the impact on trends in key market indicators such as employment, unemployment, and participation rates.

In January 2023, LFS transitioned from the National Occupational Classification (NOC) 2016 V1.3 to the NOC 2021 V1.0 for coding occupations. The differences in versions were extensive, including high-level and unit-group structural changes, creations, deletions, and combination of classification units (Statistics Canada, 2023). The most difficult to reconcile were classification units that split-off (Table 1.1).

Table 1-1
Example of NOC 2016 V1.3 to NOC 2021 V1.0 split-off

NOC ₂₀₁₆	NOC ₂₀₂₁
6541 - Security guards and related security service occupations	45100 - Student Monitors, crossing guards, and related occupations
	64410 - Security Guards and related occupations
	65329 - Other service support occupations

¹Justin Evans, Statistics Canada, 170 Tunney's Pasture Driveway, Ottawa, Ontario, Canada, K1A 0T6 (justin.evans@statcan.gc.ca); Laura Wile, Statistics Canada, 100 Tunney's Pasture Driveway, Ottawa, Ontario, Canada, K1A 0T6 (laura.wile@statcan.gc.ca)

An extensive process was required to rebase historical LFS data from the NOC₂₀₁₆ classification to the NOC₂₀₂₁ classification. Due to the volume of LFS data coded to NOC₂₀₁₆ classes that split under the NOC₂₀₂₁ classification, records from a sample of reference periods between 1987 and 2022 were selected and manually reclassified. Based on the observed NOC₂₀₂₁ split-off proportions, all other reference periods were estimated using a linear interpolation method and then assigned using a random allocation (RA) method, subject to further review by subject matter experts. As a result, respondents assigned a NOC₂₀₁₆ split-off class randomly received a NOC₂₀₂₁ class that was representative over time.

By mid-2023, some issues with the methodology used to rebase historical LFS data to the NOC₂₀₂₁ classification became apparent. Users of LFS data observed unexpected series breaks between 2023 and historical occasions. As well, due to inconsistent relationships between respondent write-ins and the assigned NOC₂₀₂₁ classes in the historical data, some unexpected NOC₂₀₂₁ classes were assigned by the production fastText model. LFS has used fastText models in their monthly coding activity since 2021 (Evans & Oyarzun, 2021), providing accurate and consistent predictions over time. However, due to the random allocation process the machine-learning model could no longer make reliable predictions on certain NOC₂₀₂₁ split-off classes; a problem that was exacerbated in NOC₂₀₁₆ classes that had high class split-off imbalances.

This paper summarizes a novel method for the LFS rebasing of text classifications using machine learning and explores the following:

- Machine-learning combined with a linear programming constraint: a methodological enhancement to current rebasing practices.
- Rebasing method selection criteria: an evaluation of the proposed method for each NOC₂₀₁₆ split-off class.
- Remodeling: assess downstream implications of the proposed rebasing method on production models.

2. Machine-Learning

2.1 Algorithm and Linear Programming

The machine-learning algorithm fastText, created by Facebook's AI Research Lab, was selected to use in the proposed text classification method. In multi-label text classification, fastText uses word embeddings— which include misspelled or made-up words, and concatenated words – to calculate the probability distribution that a text write-in belongs to each class. FastText then applies an argmax operation to identify the class corresponding to the highest probability value (Joulin et al., 2016). In a practical sense, fastText will always apply write-ins with the same text to the same class and be more likely assigned similar text write-ins to a class that is semantically similar. This is in direct contrast to a random allocation method, where the text write-in isn't considered.

To respect the proportions observed in the rebasing activity, a linear programming methodology was used to constrain fastText's probability matrix. The optimization problem, can be expressed in mathematical notation as follows:

Maximize:

$$\sum_{i=0}^n \sum_{j=1}^m x_{ij} \cdot p_{ij}$$

Subject to constraints:

$$C1: \sum_{i=1}^n x_{ij} = LC_j \text{ for } j = 1, 2, \dots, m$$

$$C2: \sum_{j=1}^m x_{ij} = 1 \text{ for } i = 1, 2, \dots, n$$

Where:

- n is the number of records.
- m is the number of labels.
- x_{ij} is a binary variable indicating whether record i is assigned label j , where $i=1,2,\dots,n$ and $j=1,2,\dots,m$.
- p_{ij} is the probability score (between 0 and 1) associated with record i for label j .
- LC_j is the required count of label j .

Stated simply, the method attempts to maximize fastText’s probability scores, assigning the most likely class to a given text write-in, while being constrained to previously assigned proportions; thus, minimizing the impact on estimate comparability.

2.2 Modeling

Models were created using NOC₂₀₁₆ split-off records manually reclassified by Statistics Canada’s Coding Center of Expertise (CCE). The authors find that while a single fastText model is used to classify NOC₂₀₂₁ in production, separate models for each of the 60 NOC₂₀₁₆ splits-off classes had an overall accuracy higher (86.8%) than a single model for all splits (71.7%). Both ML methods outperformed the baseline random allocation method (60% accuracy).

3. Rebasing Method Selection Criteria

For each of the 60 NOC₂₀₁₆ split-off classes, three different methods were compared (Table 3-1).

Table 3-1
Rebasing Methods Considered

Method	Description
RA	Maintain current historical classes, randomly assigned using proportions methodology.
ML	Use a fastText model to assign predictions to split-off records, unconstrained (top prediction assigned).
ML-LP	Use the model from ML but assign predictions using a linear programming constraint on fastText’s probability matrix to respect the proportions observed during recoding.

3.1 Metrics

Recommendations were made with the objective to improve coding at the record-level while taking into consideration apparent NOC₂₀₂₁ series breaks introduced by the RA method. Several metrics were considered (Table 3.1-1) for each NOC₂₀₁₆ series. To assess whether introducing an ML rebasing method would impact LFS published estimates, revised estimates were produced and compared to published (RA) estimates. Series breaks were identified by non-overlapping confidence intervals for two different employment estimates for the same NOC₂₀₂₁.

Series breaks were categorized based on impacting short term (number of breaks between 2022 and 2023) and long-term volatility (number of breaks year-over-year between 2013 and 2022). Nationally, there were 11/116 NOC-5 classes that had at least one series break (Table 3.1-2), and reducing to 4/116 NOC-5 classes with at least 5 months with a significant difference between the proposed series and the RA series; however, these can be attributed to low record counts or deemed justifiable when comparing to 2023 observed trends. Provincially, NOC₂₀₂₁ series breaks were mainly attributed to one NOC-5 or to small record counts. Within series, only one NOC-5 had notable month-over-month breaks; however, this was attributed to expected seasonality that was absent for some occasions for the RA series. For series where more year-over-year breaks were observed, the authors consulted with expert LFS coders to provide a recommendation. Together, these results suggest the proposed methods will have a minimal impact on published estimates.

Table 3.1-1**Metrics considered when making a rebasing recommendation**

Method	Description
Between series breaks	Favour the approach (ML or ML-LP) with the fewest breaks (ML vs. RA or ML-LP vs. RA), when compared to the RA series currently available to LFS users.
Within series breaks: historical occasions.	Favour the approach (ML, ML-LP or RA) with the fewest year-over-year and month-over-month breaks, unless breaks expected.
Proportion of records changed	Favour ML if a similar proportion of records changed by ML or ML-LP when compared to RA.
Match rate with Statistics Canada's Classification Coding System (CCS)	CCS contains thousands of job titles and the associated NOC ₂₀₂₁ class. Favour the approach (ML, ML-LP, or RA) with the highest match rate when compared to the CCS.
Consultation with coding experts	For 11/60 NOC ₂₀₁₆ classes, expert LFS coders reviewed the NOC ₂₀₂₁ classes assigned and provided a recommendation.

Table 3.1-2**National-level NOC₂₀₂₁ estimates for the last 10 years (201301 to 202306)**

NOC Level	Number of Classes	At least One Series Break (/126 Occasions, 95% CI)	Percent Break
1	9	0	0.00%
2	33	1	3.03%
3	55	1	1.82%
4	71	3	4.23%
5	116	11	9.48%

3.2 Text Write-in Comparison

The Classification Coding System (CCS) is an internal Statistics Canada tool used to assist CCE manual coders in matching respondent's write-ins to a classification class. In the hypothetical example above (Table 3.2-1), the response write-in 'school crossing guards', historically assigned to NOC₂₀₁₆6541, could be rebased to NOC₂₀₂₁64410 using the RA method as this method does not consider the text write-in. In general, an ML prediction matched more often to the CCS than a class assigned by the RA method.

Table 3.2-1**Hypothetical Text Write-in Comparison for NOC₂₀₁₆6541**

Coding Classification System Text	Coding Classification System NOC ₂₀₂₁ Class	Random Allocation (RA) NOC ₂₀₂₁ Class	Machine-Learning (ML) NOC ₂₀₂₁ Class
School Crossing Guard	45100 - Student Monitors, crossing guards, and related occupations	64410 - Security Guards and related occupations	45100 - Student Monitors, crossing guards, and related occupations
Security Guard	64410 - Security Guards and related occupations	45100 - Student Monitors, crossing guards, and related occupations	64410 - Security Guards and related occupations

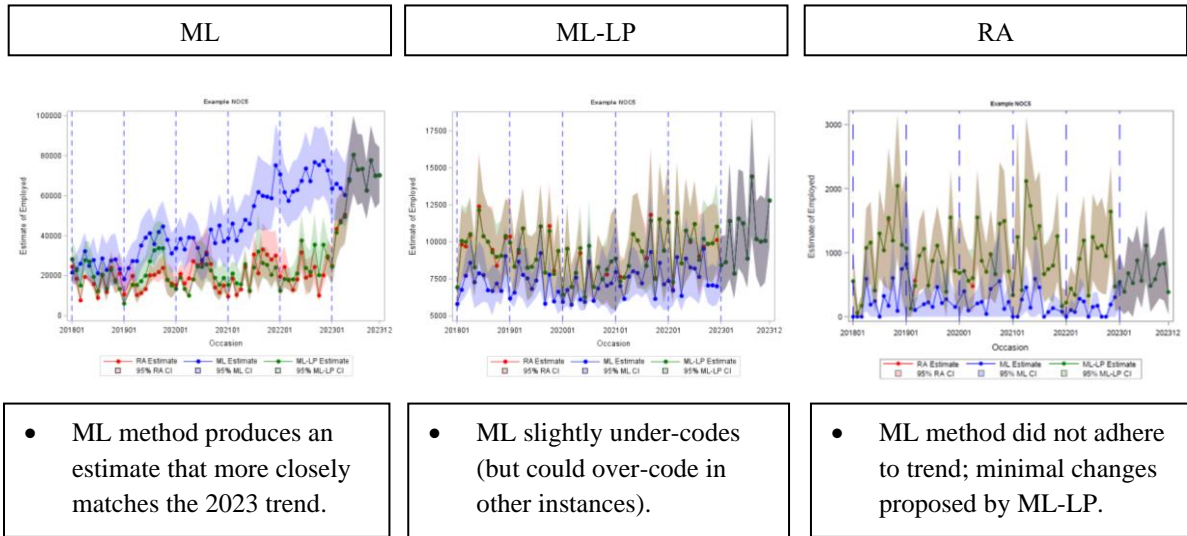
3.3 Recommendations

Each of the proposed methods proved to be useful in rebasing under specific conditions (Figure 3.3-1). The authors recommend the majority of NOC₂₀₁₆ split-off classes use a constrained or unconstrained ML rebasing method (Table 3.3-1). Machine-learning, despite its ability to harmonize similar text write-ins at the record level (ML method), had the propensity to under/over-code in high class imbalance series. Here, proportion constraints (ML-LP method) allowed corrective rebasing while adhering to published estimates. Lastly, for instances where the ML or ML-LP approach did not provide sufficient improvement, the series was left unchanged (RA method).

Table 3.3-1
NOC₂₀₁₆ split-off decision summary

Method Recommendation	NOC ₂₀₁₆ Count
ML	23
ML-LP	31
RA	6
Total	60

Figure 3.3-1
Representative time-series examples



4. Remodeling

Based on the NOC₂₀₁₆ split-off decisions, a new model was made using historical data recoded to the recommended NOC₂₀₂₁ classes to simulate the production model that would be created using the proposed methodology. The model was built using the same training and testing datasets used in the current production model (with updated NOC₂₀₂₁ classes) and validated on the most recent 6 months of LFS data. Here, it is shown that while both models have a similar overall accuracy (73.8% versus 74.9%, without using a threshold), on split-off records the rebased model outperformed (+5.3 percentage points) the production model (Table 4-1). Next, a multi-threshold per-class precision requirement was imposed to determine how many classes could be used in a production scenario. As expected, the rebased model had more classes split-off classes (+24) with at least an 85% precision or higher.

Table 4-1
Comparison of models on one year of test dataset

	Metrics	Production Model	Updated Model
Test Dataset	Overall Accuracy	73.8	74.9
	Split-off Accuracy	66.2	71.5
Multi-Threshold	Classes with 85% Precision Requirement	401	439
	Split-off Classes with 85% Precision Requirement	67	101

In a random allocation methodology, low proportion split-offs are less likely to have their NOC₂₀₂₁ class match their text write-in. In reviewing class accuracy, the current production model, which was built using the random allocation data, performs the worst on classes with low split-off proportions (Table 4-2). In contrast, classes with a high split-off proportion have relatively similar model accuracy. This highlights the importance of the rebasing activity on ensuring data quality, as it can impact downstream model decisions.

Table 4-2
Class accuracy of models on split-off classes in a 6-month validation dataset.

NOC ₂₀₂₁	Split-off Proportion	Records	Production Model Accuracy (%)	Updated Model Accuracy (%)	Difference
Class 1	0.02	38	0%	84%	+84%
Class 2	0.08	28	0%	75%	+75%
....
Class 115	0.98	13	76%	69%	-8%
Class 116	0.56	24	25%	17%	-8%

5. Summary

The ML rebasing methodology was shown to represent emerging trends, minimize time-series breaks, and harmonize historical text write-ins. The authors see that the proposed methods improve historical LFS data, which feed into downstream ML predictions that are required to be at a quality necessary for producing official statistics.

Acknowledgements

The authors would like to thank the following people for their contributions: Nadya Aliaksandrovich, Erin Howlett David Menyah, Brittany Milton, Eric Olson, Sylvia White, Cynthia Breault, Alexander Calvano, Benoit Durand, Brandon Gould, Johanne Madore, Derek Strelieff, Eric Laflamme, Julie Portelance, Kerry Schneider, Jessica Mulligan, Javier Oyarzun, Michelle Simard, and Yi Li.

References

- Evans, J.J, and Oyarzun, J. (2021), “Need for Speed: Using fastText (Machine Learning) to Code the Labour Force Survey”, *Proceedings of Statistics Canada Symposium 2021*. Accessed from <https://www150.statcan.gc.ca/n1/en/catalogue/11-522-X202100100013>
- Statistics Canada (2023), “Introduction to the National Occupational Classification (NOC) 2021 Version 1.0”, Retrieved from <https://www.statcan.gc.ca/en/subjects/standard/noc/2021/introductionV1#a8>.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2016), “Bag of Tricks for Efficient Text Classification”,
<https://arxiv.org/pdf/1607.01759>.