Evaluación de técnicas de procesamiento de lenguaje natural y *Machine Learning* **para los procesos de codificación de encuestas en hogares**

Evaluation of Natural Language Processing and Machine Learning Techniques **for Household Survey Coding Processes**

José Alejandro Ruiz Sánchez,* Jael Pérez Sánchez y Adrián Pastor López Monroy*****

* Instituto Nacional de Estadística y Geografía (INEGI), jose.ruizs@inegi.org.mx
** INEGI, jael.perez@inegi.org.mx
*** Centro de Investigación en Matemáticas, A. C., pastor.lopez@cimat.mx

Artificial intelligence (AI), machine learning and modern computer technologies concepts. Business, Technology, Internet and network concept./ putilich/ iStock

De los múltiples procesos productivos llevados a cabo dentro de las oficinas nacionales de estadística se encuentra el de codificación, el cual consiste en la asignación automática o manual de claves alfanuméricas a un registro u observación. Este mapeo a un conjunto de categorías predefinidas permite agrupar registros bajo una misma descripción, lo cual facilita su manejo y análisis. Un porcentaje importante de estas tareas de codificación se realizan con ayuda de algoritmos determinísticos basados en reglas de decisión; sin embargo, otros procesos utilizan en mayor medida la asistencia de expertos humanos. El trabajo que a continuación presentamos tiene por objetivo valorar el uso e incorporación de técnicas de procesamiento de lenguaje natural (PLN) y de *Machine Learning (ML)* para incrementar el porcentaje de registros clasificados de manera automática. Para ello, tomamos las variables de *ocupación* y *actividad económica* de la Encuesta Nacional de Ingresos y Gastos de los Hogares 2018. Los resultados obtenidos muestran que sería posible trasladar 50 % de los registros que actualmente se codifican con asistencia humana hacia un proceso de codificación automatizada con algoritmos de PLN y *ML*.

Palabras clave: procesamiento de lenguaje natural; *Machine Learning;* codificación; ENIGH.

Recibido: 26 de julio de 2021.
Aceptado: 27 de octubre de 2021.

National Statistic Offices carry out multiple production processes, coding being one of them. Coding is referred to the assignment of alphanumeric keys to a particular observational unit. The coding process can be either automatic or manual and it is based on certain additional information. This mapping to a set of predefined categories allows grouping records under the same description, which facilitates its management and analysis. Currently a great percentage of the coding tasks are made by deterministic algorithms or decision rules. However, there are processes where human intervention to code is still largely required. The paper we present assesses the use and incorporation of Natural Language Process (NLP) and Machine Learning (ML) to increase the percentage of automatically coded records. We evaluate the process on two variables from the National Survey of Household Income and Expenditure (ENIGH by its acronym in Spanish) 2018: *occupation* and *economic activity*. Our results show it could be possible to transfer 50% of the records coded by humans to be automatically coded by NLP and ML.

Key words: Natural Language Process; Machine Learning; Coding; ENIGH.

Introducción

En todo proyecto de generación de información estadística por lo general existen preguntas abiertas que dan como resultado un conjunto de descripciones (usualmente en forma de texto) sobre la temática que se está levantando. Al proceso de asignarle una clave alfanumérica a estas con fines de explotación de la información se le llama codificación. En los censos de población y vivienda, así como en las encuestas en hogares, hay diferentes variables que entran al proceso de codificación, por ejemplo: parentesco, lengua indígena, religión, lugar de nacimiento, lugar de residencia, ocupación de la persona o la actividad económica en la cual trabaja.

Actualmente, el área del Instituto Nacional de Estadística y Geografía (INEGI) a cargo de la codificación de encuestas en hogares, como la Encuesta Nacional de Ingresos y Gastos de los Hogares (ENIGH), realiza el proceso en dos etapas: la primera se le llama automática porque se hace con un conjunto de

reglas determinísticas que son aplicadas a través de algoritmos computacionales, donde se conjuga con un preprocesamiento de los textos para lograr una mayor cantidad y calidad de registros codificados de manera correcta; la segunda se denomina codificación asistida, pues es realizada por humanos a través de un sistema que facilita dicho trabajo. Tanto para la codificación automática como para la asistida, se garantiza un nivel de calidad mínimo aceptable que está en función de la complejidad de cada variable. Para la mayor parte de las variables, la calidad de la codificación se encuentra establecida en un nivel mínimo de 98 %, no así para *ocupación y actividad económica*, donde clasificar las descripciones resulta mucho más complejo que para el resto de variables.

Para las variables de *ocupación* y *actividad económica*, la calidad oscila entre 90 y 94 %, dependiendo del proyecto que se trate, sea censo o encuestas en hogares, con un porcentaje de codificación automática entre 70 y 76; el complemento se hace de manera asistida. La verificación de la calidad de la codificación se hace de forma implícita en el proceso; para la automática se emplea un muestreo probabilístico para cada lote de información y para cada estrategia de codificación, mientras para la asistida, se emplea un muestreo de aceptación por lotes (tal cual se hace en la industria) donde, para cada carga de trabajo hecha por cada codificador se obtiene una muestra; si esta no pasa el umbral de calidad mínimo aceptable, se vuelve a codificar la carga completa (ver *Anexo*).

El trabajo que a continuación presentamos tiene por objetivo valorar el uso e incorporación de técnicas de procesamiento de lenguaje natural (PLN) y de *Machine Learning* (*ML*) para incrementar el porcentaje de registros clasificados de manera automática. Para ello, tomamos las variables de *ocupación* y *actividad económica* de la ENIGH 2018.

El documento está dividido en dos secciones: en la primera reportamos el proceso metodológico que seguimos, así como los principales resultados, y en la segunda proponemos la operacionalización del proceso metodológico en conjunción con los actuales procesos productivos.

1. Procesamiento de lenguaje natural para codificación automática

A través de los años, el INEGI ha desarrollado procesos de codificación para distintas variables, lo cual ha resultado en una gran cantidad de información etiquetada por personal especialmente capacitado para tales tareas. Sin embargo, para algunas de las variables, el proceso de codificación sigue representando un reto en términos de tiempo, recursos humanos y mejora en calidad. Los avances recientes en el campo del PLN y *ML*, junto con los insumos acumulados por las oficinas nacionales de estadística (ONE), ofrecen la posibilidad de incorporar nuevas metodologías y así obtener mejores resultados en términos de calidad y eficiencia. Por PLN nos referimos al conjunto de técnicas y algoritmos empleados para el tratamiento automático del lenguaje (Jurafsky and Martin, 2008) y en nuestro caso, al procesamiento, modelación y representación del texto contenido en la base de datos en cuestión. Para la parte de *ML*, nos enfocamos en clasificadores automáticos basados en procesos de optimización, que parametrizan modelos mediante el insumo de datos para descubrir relaciones y patrones relevantes entre ellos (Svensén and Bishop, 2007).

En esta sección elaboramos una primera aproximación a las técnicas de PLN aplicadas a las tareas de clasificación automática para las variables *ocupación* y *actividad económica* que reportan los informantes de la ENIGH. Nuestro objetivo es explorar la viabilidad del uso de estos métodos dentro de alguna de las fases del proceso de codificación.

1.1 Datos

El ejercicio lo realizamos sobre la ENIGH 2018, cuyas variables de *ocupación* y *actividad económica* ya están codificadas según los procesos actuales del INEGI. La base de datos tiene 158 568 registros etiquetados. Los campos descriptivos con base en los cuales se realizó la codificación son: *Nombre de la ocupación, Tareas o funciones, Activi-*

dad económica de la empresa o institución y *Nombre de la empresa*. Adicionalmente, se integran las siguientes características de apoyo: *Lugar donde realizó las actividades*, *Clasificación de la empresa*, *Si tiene personal a su cargo*, *Nivel académico*, *Grado académico*, *Nivel académico aprobado*, *Grado aprobado*, *Tamaño de la empresa*, *Trabaja dentro del país* y *Pregunta complementaria a nombre de la empresa*.

1.2 Procesamiento y resultados

En términos generales, el desarrollo metodológico que hemos seguido para la aplicación de las técnicas de PLN y *ML* consta de tres fases: 1) preprocesamiento de los datos, 2) vectorización numérica de los textos y 3) clasificación a través de algoritmos de aprendizaje supervisado[1] de *ML*.

1.2.1 Preprocesamiento

Como se comentó en la sección anterior, el proceso actual de codificación que se realiza en la ENIGH se compone de dos etapas, aquella basada en algoritmos determinísticos de reglas de decisión y otra donde interviene personal especializado. Para la primera, es especialmente importante contar con un texto estandarizado; para ello, se realizan correcciones ortográficas, lematizaciones y truncado de palabras. Este mismo proceso es el que hemos seguido como paso inicial para el tratamiento de los datos para el uso de PLN y *ML*. Una vez estandarizado el texto, generamos conjuntos de *n*-gramas (partición de una palabra o frase en subconjuntos de igual cardinalidad; así, existen *n*-gramas a nivel carácter y palabra. Este último consiste en la agrupación de palabras secuenciales; así, un *n*-grama a nivel palabra de cardinalidad o longitud 2 resulta de la unión de dos palabras secuenciales), los cuales serán el insumo para la etapa de vectorización. Los *n*-gramas usualmente benefician la efectividad de clasificación debido a que capturan información del contexto o conceptos compuestos por *n* palabras.

1 Los métodos de *ML* supervisado establecen relaciones entre un conjunto de covariables con otra variable objetivo, la cual es de interés predecir o estimar.

1.2.2 Vectorización

En términos generales, esta etapa consiste en representar numéricamente un texto, el cual puede estar compuesto por un conjunto de caracteres alfanuméricos (*tokens*) y, por lo tanto, contener *n*-gramas o palabras completas.

Uno de los métodos tradicionales en PLN para vectorización consiste en contabilizar la repetición de cada uno de los *tokens* que aparece en un documento (bolsa de palabras). Estos pueden ser palabras, *n*-gramas o cualquier otra unidad simbólica que se determine de manera previa. Algunas de sus variantes surgen de agregar ponderaciones a este conteo, tal es el caso de la técnica *Term frequency-Inverse document frequency* (*TF-IDF*).

Consideremos N, número de documentos (o registros), cada uno de los cuales está compuesto por una cantidad finita de términos o *tokens*. Cada término i puede ser representado por un vector numérico de dimensión menor o igual a N, de acuerdo con la siguiente fórmula:[2]

$$w_{i,j} = (1 + log\ (tf_{i,j})) * log\ log\ \left(1 + \frac{N}{(1 + df_i)}\right),$$

donde:

$tf_{i,j}$ = número de veces que aparece el término i en el documento j.
df_i = número de documentos que contienen el término i.
N = número de documentos.

Por ejemplo (ver ilustración), consideremos los siguientes tres ($N = 3$) documentos (textos) obtenidos de la ENIGH. En este caso la palabra *FRUTA* aparece en ocho ocasiones y solo en uno de los textos ($df_i = 1$).

2 Para el procesamiento de la técnica *TF-IDF,* usamos el paquete *Keras* en *Python*. La fórmula que se muestra fue tomada de la siguiente liga: https://github.com/keras-team/keras-preprocessing/blob/master/keras_preprocessing/text.py

Ilustración

	FRUTA	ALBAÑIL	DEPARTAMENTO	PISO	AGRICULTURA	…
1. "ALBAÑIL PEGAR PISO HACER ACABAR COMO REPELLAR PEGAR PISO *ETC*. DEPARTAMNETO NO DEPARTAMENTO"	0	1	2	2	0	…
2. "AYUDANTE DE ALBAÑIL BATIR MEZCLA ACARREAR LADRILLO QUEHACER COSA DE LADRILLO"	0	1	0	0	0	…
3. "AGRICULTOR CHAPEAR SU AGRICULTURA DE FRUTA FRUTA FRUTA FRUTA EN UNA PARCELA AGRICULTURA DE FRUTA FRUTA FRUTA FRUTA EN UNA PARCELA"	8	0	0	0	2	…

↓

	FRUTA	ALBAÑIL	DEPARTAMENTO	PISO	AGRICULTURA	…
1. "ALBAÑIL PEGAR PISO HACER ACABAR COMO REPELLAR PEGAR PISO ETC. DEPARTAMENTO NO DEPARTAMENTO"	0	1**log*(3/2)	2**log*(3/1)	2**log*(3/1)	0	…
2. "AYUDANTE DE ALBAÑIL BATIR MEZCLA ACARREAR LADRILLO QUEHACER COSA DE LADRILLO"	0	1**log*(3/2)	0	0	0	…
3. "AGRICULTOR CHAPEAR SU AGRICULTURA DE FRUTA FRUTA FRUTA FRUTA EN UNA PARCELA AGRICULTURA DE FRUTA FRUTA FRUTA FRUTA EN UNA PARCELA"	8**log*(3/1)	0	0	0	2	…

De esta manera, cada uno de los documentos (renglones) en la ilustración es representado por un vector numérico de una dimensión finita. De forma intuitiva, la técnica castiga términos que aparecen en todos los documentos y, por lo tanto, suelen ser un tanto discriminativos. Naturalmente, este proceso puede ser fácilmente extendido cuando se incluyen *n*-gramas a nivel carácter y/o palabra.

Una vez que obtenemos una representación vectorial para cada documento (en nuestro caso, un documento está conformado por la concatenación de las respuestas textuales de un informante de la ENIGH convertidas en *n*-gramas), los resultados pasan a la siguiente etapa, que es la de clasificación automática a través de algoritmos de *Machine Learning*.

1.2.3 Resultados de la codificación automática

Los resultados de los distintos ejercicios que se presentan a continuación tienen como finalidad aportar información sobre las variaciones metodológicas y parametrizaciones que resultaron favorables para nuestros problemas de clasificación (*ocupación* y *actividad económica* para la ENIGH).

De los 158 568 registros originales en nuestra base de datos, excluimos 58 pertenecientes a clases (códigos) con menos de cuatro registros. De los datos restantes, usamos 75 % de ellos como conjunto de entrenamiento (manteniendo las proporciones de las categorías) y el restante 25 % como conjunto de prueba. Los ejercicios realizados fueron para clasificar, por separado, la variable *ocupa-*

ción conformada por 461 clases, y la de *actividad económica* que cuenta con 157.

Con respecto a los algoritmos de *ML*, para este proyecto empleamos los tradicionales de aprendizaje supervisado (donde existe una variable a predecir y un conjunto de predictores), como *Support Vector Machine* (*SVM*), regresión logística, *Radom Forest*, *K Nearest Neighbors* (*K-NN*), *Naive Bayes*, entre otros (Chih-Chung y Chih-Jen, 2019; Platt, 1999; Hsiang-Fu *et al*., 2011; Breiman, 2001; Zhang, 2004). De estos, el que obtiene en general resultados más sobresalientes es el *SVM*, el cual es un algoritmo de aprendizaje automático que tiene el objetivo de encontrar un hiperplano óptimo que separe las instancias que pertenecen a dos diferentes clases (Svensén and Bishop, 2007). De manera específica, y sea $\{\boldsymbol{X}_i, y_i\}$ el conjunto de pares instancia-categoría de ejemplos de entrenamiento, donde $\boldsymbol{X}_i \in \mathbb{R}^d$ $y \in \{-1, +1\}$ con d dimensionalidad del problema (e.g., el tamaño del vocabulario). El *SMV* trata de determinar un mapeo de los ejemplos de entrenamiento a las categorías objetivo por medio de la utilización de la siguiente función lineal:

$$f(x) = sign(\sum_i \alpha_i y_i k(x_i, \boldsymbol{x}) - b)$$

donde α_i y y_i son los pesos y etiqueta del ejemplo de entrenamiento i. Para mapear los vectores de entrada (x_i, x_j) al espacio de características, se utiliza la función de kernel $k(x_i, \boldsymbol{x})$. De manera intuitiva, $k(x_i, \boldsymbol{x})$ mide la similitud entre las instancias x_i y x_j. Los parámetros α y b son aprendidos por medio de optimización, y la selección de la función de kernel es de vital importancia (Svensén and Bishop, 2007).

De entre los algoritmos evaluados, el que mejores resultados arrojó fue el *SVM*. En próximas iteraciones de este proyecto incluiremos algoritmos en el estado del arte para valorar su factibilidad.

Excepto cuando se especifique lo contrario, en los ejercicios usamos las mismas variables predictoras como insumo para la clasificación, las cuales se enlistan en el cuadro 1.

El primer conjunto de ejercicios tuvo como finalidad probar la utilidad de segmentar una palabra, combinar segmentos de palabras o hacer uso de palabras completas en las variables de texto; es decir, experimentamos con diversas combinaciones de *n*-gramas, tanto en longitud como uniones entre ellos. En primer lugar, cada uno de los registros de texto los seccionamos a nivel carácter en *n*-gramas de longitud 6 (en diversos ejercicios probamos distintas longitudes y esta fue la que mejores resultados arrojó). Después, se escogieron los 30 mil *n*-gramas más frecuentes (en diversos ejercicios

Cuadro 1

Variables (características o *features*) usadas para la clasificación

Principales (texto)	Auxiliares (categóricas)
Nombre de la ocupación	Lugar donde realizó las actividades
Tareas o funciones	Clasificación de la empresa
Actividad económica de la empresa o institución	Si tiene personal a su cargo
Nombre de la empresa	Nivel académico
	Grado académico
	Nivel académico aprobado
	Grado aprobado
	Tamaño de la empresa
	Trabaja dentro del país
	Pregunta complementaria a nombre de la empresa

probamos con distintas dimensiones y esta fue la que mejores resultados arrojó) para conformar una matriz de 30 mil columnas a través del proceso *TD-IDF*, cuyo resultado lo empleamos dentro de un algoritmo de *SVM* para clasificar automáticamente los registros del conjunto de prueba de la ENIGH (las métricas reportadas a lo largo de esta sección son: *accuracy*, precisión, *recall* y F1; sin embargo, los ejercicios fueron guiados tomando en consideración el resultado del *accuracy*, el cual representa la proporción de registros cuyo código asignado por el proceso actual del INEGI coincide con aquel código asignado por el algoritmo de *ML*). El resultado de este acercamiento inicial se muestra en el primer renglón del cuadro 2. El segundo ejercicio incorpora tanto *n*-gramas a nivel carácter de longitud 6 como de 10; para cada uno de estos conjuntos, generamos una matriz *TF-IDF* de 30 mil columnas, que al concatenarlas obtenemos una representación vectorial de 60 mil para cada registro de nuestra base de datos. El tercer y último ejercicio que reportamos en el cuadro 2 resulta de generar una matriz *TF-IDF* con *n*-gramas a nivel carácter de longitud 6 y *n*-gramas a nivel palabra de cardinalidad 2 (es decir, un *n*-grama está conformado por la agrupación de secuencia de palabras completas; en este caso, se crean pares de palabras). Posterior a la aplicación de la técnica *TF-IDF*, y antes de la implementación de los algoritmos de *ML*, realizamos un proceso de normalización con la norma L2 para las matrices resultantes *TF-IDF* (el proceso de normalización se hace sobre matrices concatenadas).

Como se muestra en el cuadro 2, este último ejercicio fue el que mejor resultado arrojó en términos de *accuracy* (0.8793).

Con el segundo conjunto de ejercicios (ver cuadro 3) evaluamos la conveniencia de que la asignación de un código a una determinada observación esté basada no solo en el resultado de un algoritmo, sino en la combinación de resultados de distintos algoritmos (ensamble). Para nuestras tareas de codificación, los mejores resultados (en términos del *accuracy*) los obtuvimos al clasificar los registros de la ENIGH a través de un ensamble con los siguientes métodos: *SVM*, regresión logística, *Random Forest*, redes neuronales, *XGBoost, K-NN*, *Naive Bayes* y árboles de decisión. También, encontramos que la mejora es significativa usando palabras completas.

Cuando obtenemos los resultados individuales de cada algoritmo de clasificación, observamos que algunos producen clasificaciones sustancialmente más certeras, por lo que sería razonable en uno de ensamble otorgar mayor peso a aquellos que en lo individual lo hacen mejor. Haciendo la analogía con un sistema de votación, vamos a dar más votos a la clasificación que nos indican aque-

Cuadro 2

Accuracy para distintas representaciones del texto. Clasificador _SVM_

	Actividad económica	Ocupación
6-gramas (carácter)	0.8782	0.8204
6-gramas (carácter), 10-gramas (carácter)	0.8781	0.8189
6-gramas (carácter), 2-gramas (palabra)	0.8793	0.8188

Cuadro 3

Accuracy para distintas representaciones del texto. Clasificador ensamble

	Actividad económica	Ocupación
6-gramas (carácter)	0.8849	0.8474
6-gramas (carácter), 10-gramas (carácter)	0.8825	0.8467
6-gramas (carácter), 2-gramas (palabra)	0.8905	0.8505

llos algoritmos que tuvieron un desempeño mejor. Dado un registro por clasificar, aquel código que tenga más votos será el que sea asignado a este. Siguiendo esta idea, encontramos una mejora en el *accuracy* si otorgamos un número de votos distinto a cada algoritmo del ensamble. El número de votos asignado a cada algoritmo fue obtenido guiándonos por el desempeño y los resultados obtenidos: cuatro a *SVM*, dos a regresión logística, uno a *Random Forest*, tres a redes neuronales, dos a *XGBoost*, uno a *K-NN*, uno a *Naive Bayes* y tres a árboles de decisión. Las mejoras son notorias en métricas como *Recall* y F1 (ver cuadro 4).

Para la variable de *ocupación* (ver cuadro 5), los pesos que mejores resultados arrojaron fueron los siguientes: cuatro a *SVM*, dos a regresión logística, dos a *Random Forest*, cuatro a redes neuronales, tres a *XGBoost*, uno a *K-NN*, uno a *Naive Bayes* y tres a árboles de decisión.

Todos los resultados anteriores fueron calculados considerando una sola fase en el proceso de clasificación (como es común); es decir, independientemente del número de clases, todos los registros se intentan clasificar en alguna de ellas y en una sola etapa. Sin embargo, una de las características de los códigos de *ocupación* y *actividad económica* es que son jerárquicos. Por ejemplo, para el caso de *actividad económica*, cada código está compuesto por cuatro dígitos: los dos primeros corresponden al sector al que pertenece el código, de tal manera que varios de estos pueden pertenecer al mismo sector. La variable de *actividad económica* está conformada por 157 clases que pertenecen a 25 sectores y *ocupación* tiene 461 agrupadas en 52 sectores. Esta estructura en los códigos puede ser aprovechada para intentar aumentar el poder predictivo del algoritmo.

Para probar esta idea, realizamos el siguiente experimento: al conjunto de entrenamiento adicionamos un grupo de variables dicotómicas dentro de las predictivas; cada una de las dicotómicas es marcada con 1 si el registro pertenece a ese sector y 0 si es lo contrario. Este nuevo grupo de variables adicionales, junto con las del cuadro 2, las usamos para entrenar y obtener un algoritmo integral de clasificación (*modelo clase*) basado en los registros del conjunto de entrenamiento. Para el conjunto de prueba, y dado que no debemos incorporar directamente el sector al que pertenece cada registro, adicionamos una etapa para clasificar y pronosticar el sector, esto lo logramos entrenando un modelo basado en registros del conjunto de entrenamiento y con las variables del cuadro 2 (*modelo sector*); de esta manera, para el conjunto de prueba obtenemos, en primer lugar, un sector pronosticado que transformamos en variable dicotómica; después,

Cuadro 4

Resultados usando n-gramas de seis caracteres y de dos palabras. Actividad económica

	Accuracy	Precisión	*Recall*	F1
Ensamble con mismos pesos	0.8905	0.6925	0.6149	0.6365
Ensamble con pesos diferenciados	0.8921	0.6767	0.6420	0.6512

Cuadro 5

Resultados usando n-gramas de seis y 10 caracteres. Ocupación

	Accuracy	Precisión	*Recall*	F1
Ensamble con mismos pesos	0.8447	0.6441	0.5384	0.5639
Ensamble con pesos diferenciados	0.8505	0.6437	0.5637	0.5831

usamos esta clasificación sectorial para ordenar las clases (códigos) basado en el *modelo clase*.

Los resultados de este ejercicio se muestran en los cuadros 6 y 7. Para comparar, en el primer renglón replicamos los resultados mostrados en el cuadro 2, donde empleamos *SVM* y matrices *TF-IDF* con *n*-gramas a niveles carácter de longitud 6 y palabra de cardinalidad 2. Usando *SVM* y con la misma arquitectura en la matriz *TF-IDF*, observamos que el ejercicio basado en dos etapas (primero sector y luego clase) para la variable de *actividad económica* mejora solo en la métrica de precisión cuando clasificamos *actividad económica* (ver cuadro 6); sin embargo, para *ocupación* parece haber una mejora importante tanto en *accuracy* como en precisión (ver cuadro 7).

2. Incorporación de algoritmos de *ML* dentro del proceso de codificación en el INEGI

Un tema central para las ONE es asegurar la correcta incorporación de nuevos procedimientos a la producción corriente. En esta sección presentamos una propuesta para adecuar el proceso actual de codificación en las encuestas en hogares que realiza el INEGI e incorporar técnicas de PLN que ayuden a aminorar la carga en la codificación asistida.

Como referencia, si solo empleáramos PLN para codificar el 100 % de los registros, obtendríamos un *accuracy* de 89.2 % para la variable de *actividad económica* y de 85 % para la de *ocupación* (recordemos que el *accuracy* se refiere a la proporción de observaciones o registros cuyo código original coincide con aquel asignado por el algoritmo de *ML*). Sin embargo, otra opción es codificar con el algoritmo de *ML* solo aquellos registros que pertenecen a clases cuyo *accuracy* por clase (*accuracy* interno) es superior a determinado umbral (por ejemplo, mayor a 95 %). Este ejercicio lo resumimos en la gráfica 1. Como se observa, si deseamos un *accuracy* de 95 %, el algoritmo lograría clasificar cerca de 65 % del total de registros en la base de datos de la ENIGH, al excluir aquellas clases cuyo *accuracy* individual no pasó el umbral.

El escenario anterior no considera la coexistencia de los procesos actuales de codificación con una metodología de PLN. En la sección 1 comentamos

Cuadro 6

Modelo de clasificación jerárquica usando *SVM*. Actividad económica

	Accuracy	Precisión	*Recall*	F1
Una sola etapa (6-gramas carácter, 2-gramas palabras)	0.8793	0.6372	0.6760	0.6511
Dos etapas (6-gramas carácter, 2-gramas palabras)	0.8774	0.6600	0.6452	0.6443

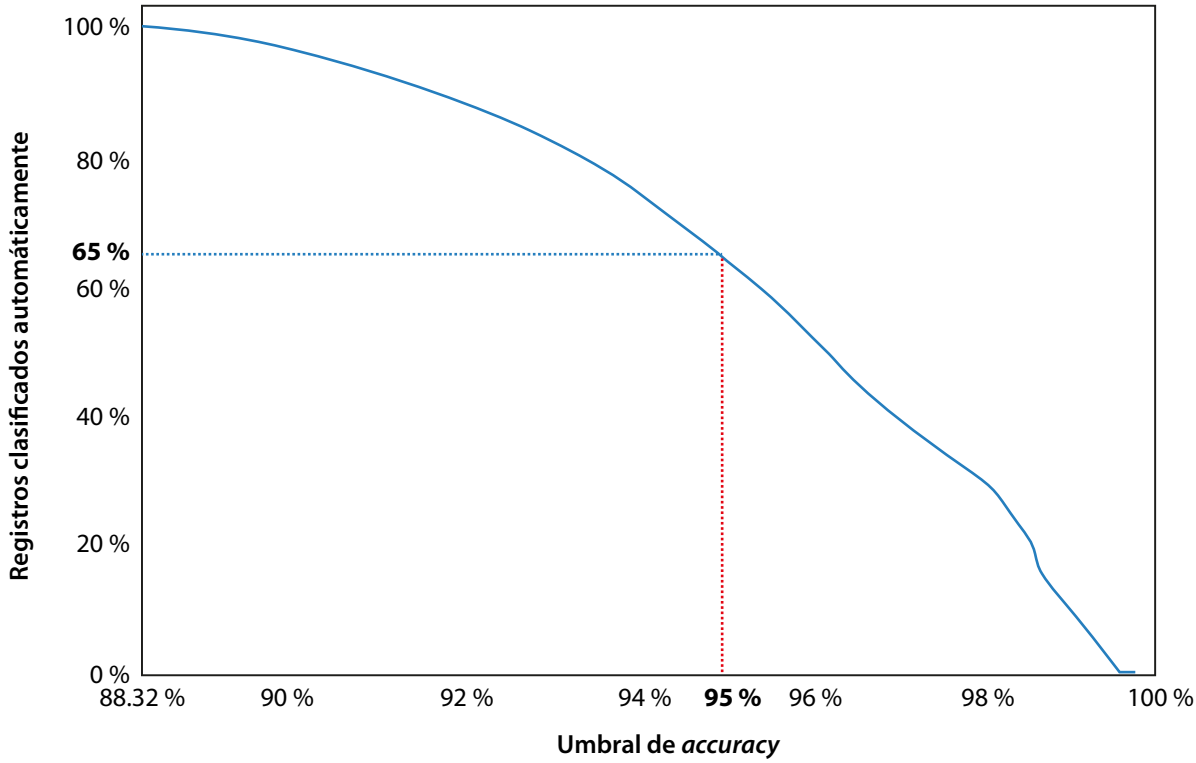
Cuadro 7

Modelo de clasificación jerárquica usando *SVM*. Ocupación

	Accuracy	Precisión	*Recall*	F1
Una sola etapa (6-gramas carácter, 2-gramas palabras)	0.8188	0.5353	0.5918	0.5531
Dos etapas (6-gramas carácter, 2-gramas palabras)	0.8312	0.5786	0.5730	0.5648

Gráfica 1

Umbral de *accuracy* por clase vs. porcentaje de registros clasificados automáticamente

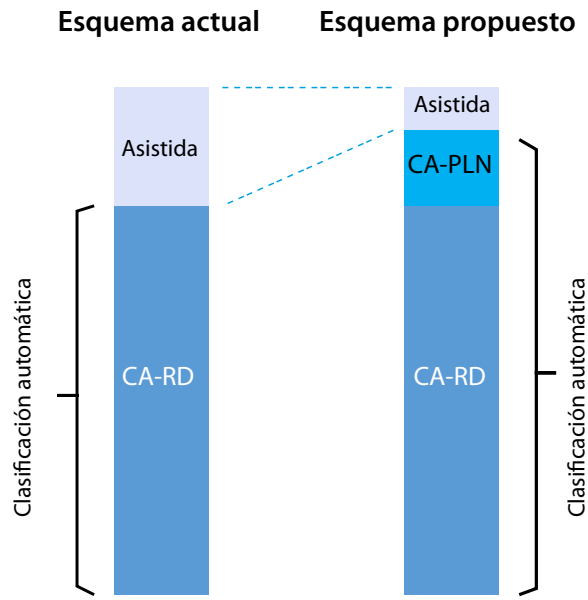


sobre el proceso actual, el cual está dividido en dos etapas: en la primera se emplea un algoritmo de clasificación automática con reglas determinísticas (CA-RD) y en la segunda se utilizan codificadores humanos (codificación asistida) para aquellos registros que no se logró captar en la etapa inicial. Nuestra propuesta consiste en la incorporación, como fase intermedia, de la metodología desarrollada en secciones previas; es decir, una vez aplicado CA-RD, usamos el algoritmo de PLN (CA-PLN) para un subconjunto de registros que no pudieron ser clasificados empleando CA-RD. Así, la clasificación automática sería la unión de CA-RD y CA-PLN. Los registros restantes se codificarían por humanos, reduciendo así, la carga de trabajo (ver figura).

Ahora bien, ¿cuáles registros que anteriormente eran clasificados por humanos se tratarían ya usando *ML*? La primera posibilidad es confiar en el algoritmo de *ML* para sustituir en su totalidad la codificación asistida; de esta forma, el 100 % de los

Figura

Propuesta operativa para el uso de PLN para las tareas de codificación



registros serían clasificados automáticamente (CA-RD + CA-*ML*). El costo de esta decisión recaería en la calidad de la clasificación (coincidencia entre la clasificación asistida y la clasificación del algoritmo de *ML*, es decir, el *accuracy*): si tomamos como *ground-truth* la clasificación de la codificación asistida y esta la comparamos con el código que sería asignado usando CA-*ML*, el porcentaje de coincidencia sería solo de 68.6.

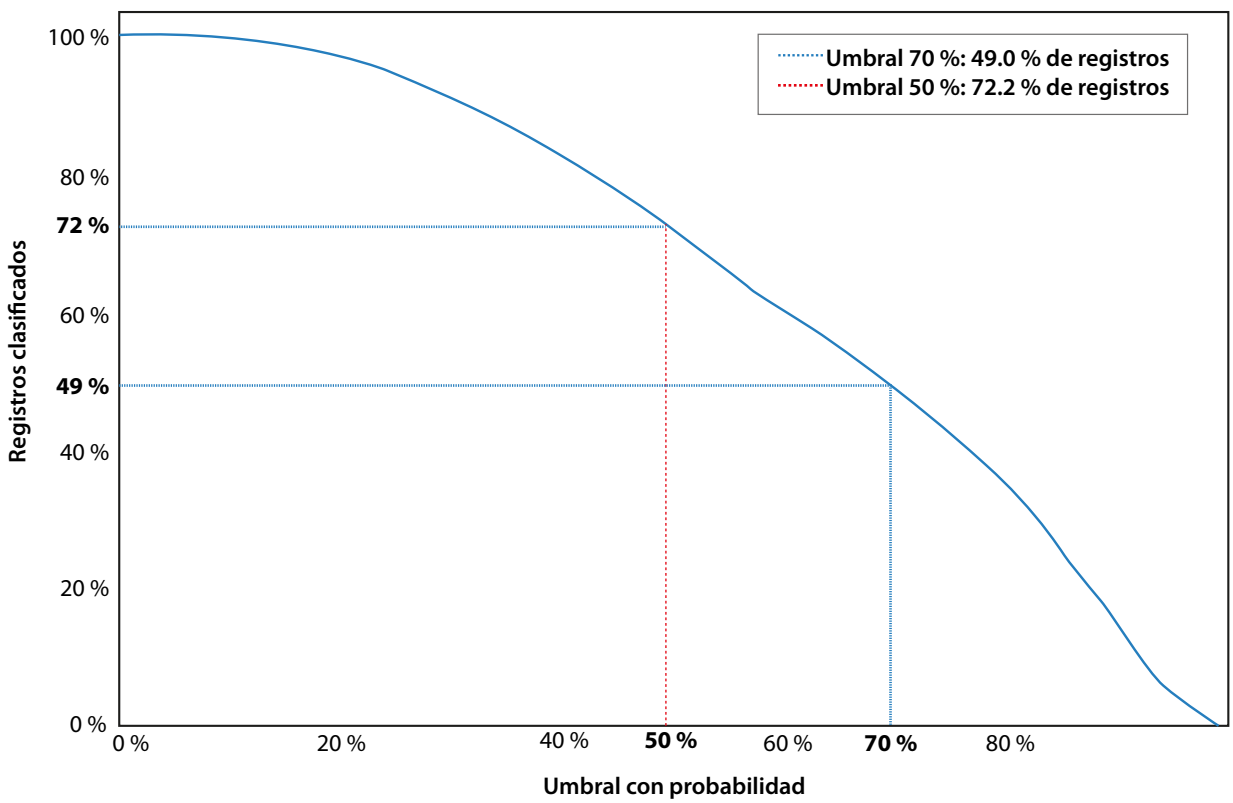
Otra posibilidad es considerar solo un subconjunto de los registros clasificados de forma asistida. Para ello, proponemos emplear una métrica de certidumbre (similar a la probabilidad) que viene asociada con los algoritmos de *ML*. En la implementación de estos se puede obtener el siguiente par de valores para cada registro: código asignado y un grado de certidumbre asociado a ese código. La idea intuitiva para su empleo recae en su correlación positiva con la coincidencia entre el algoritmo de *ML* y la parte asistida: a mayores valores de la métrica de certidumbre, mayor será la probabilidad de que coincidan. Esta métrica esta normalizada y toma valores entre 0 y 1.

El proceso que se propone para los registros que no pueden ser clasificados con reglas determinísticas es el siguiente: a) seleccionar un umbral de certidumbre (por ejemplo, 0.7), b) asignar el código pronosticado por el algoritmo de *ML* solo a los registros que tengan un valor de certidumbre igual o mayor a ese umbral y c) los registros restantes serían clasificados de forma asistida. De esta forma, tendremos un esquema como el de la figura arriba presentada.

Entre mayor es el valor del umbral elegido, mayor es el porcentaje de coincidencias entre lo que clasifican los humanos y lo que dice el algoritmo de *ML*; sin embargo, entre mayor es el umbral, menor es el número de registros que pueden ser clasificados con el algoritmo de *ML*. Por ello, el umbral deberá estar en función del nivel mínimo aceptado de coincidencias (*accuracy*). Este *trade-off* se presenta en la gráfica 2.

Gráfica 2

Trade-off entre los registros codificados por _ML_ y el umbral de certidumbre



Así, para el caso de actividad económica, si seleccionamos un umbral de 0.7, el porcentaje de coincidencias con respecto a la codificación asistida es de 87.7 y el de registros que pasan el umbral respecto al total de registros asistidos, de 49; en otras palabras, la carga de trabajo para los humanos se reduciría casi a la mitad, y el porcentaje de coincidencia entre el código que ellos asignarían con el que asignaría el algoritmo sería de 87.7. Si el umbral se incrementa a 0.8, el valor de coincidencias aumenta a 91.7 %, pero la carga de trabajo para los humanos se reduce a 34.7 por ciento.

Otro posible uso de los algoritmos de *ML* dentro de los procesos actuales de codificación es para identificar potenciales errores humanos. Para ello, se aprovecharía la distribución de probabilidades que un algoritmo de *ML* genera para alertar de posibles discrepancias entre lo que pronostica un algoritmo con alta probabilidad y la etiqueta que otorga un codificador humano. Este proceso puede ser especialmente de utilidad en proyectos donde se contratan de forma eventual codificadores humanos y que, por lo general, tienen poca experiencia, por ejemplo, en los censos de población y vivienda.

3. Conclusiones

Las oficinas nacionales de estadística llevan a cabo tareas diarias cuyas características las hace propensas al aprovechamiento de los avances metodológicos y tecnológicos, tal es el caso de los procesos de codificación, donde, a partir de un conjunto de covariables auxiliares, se determina uno de los códigos que le será asignado a un registro en una base de datos. Estas covariables pueden ser de texto, categóricas o numéricas.

El objetivo de este documento es evaluar la factibilidad de incorporación de técnicas de procesamiento de lenguaje natural dentro de los actuales procesos productivos que se siguen para la codificación de variables de encuestas en hogares realizadas por el INEGI. Una de ellas es la ENIGH, y entre las variables que se codifican para esta se encuentran las de *ocupación* y *actividad económica*. Actualmente, estas variables requieren de una cantidad considerable de recursos humanos debido a la dificultad de automatización con técnicas convencionales.

La propuesta elaborada en este trabajo permitiría reducir la carga a los codificadores humanos al incorporar PLN para automatizar la asignación de códigos para las variables de *ocupación* y *actividad económica*. De acuerdo con las pruebas que llevamos a cabo, se podrían codificar automáticamente 50 % de los registros que en la actualidad se hacen de forma manual y obtener porcentajes de coincidencia (*accuracy*) cercanos a 90 por ciento.

Estos resultados, a su vez, constituyen un punto de referencia para potenciales mejoras en los algoritmos evaluados, toda vez que los avances en esta área están en constante desarrollo y evolución.

Fuentes

Breiman, L. "Random Forest", en: *Machine Learning*. 45, 2001, pp. 5-32.

Chih-Chung, C. y L. Chih-Jen. *LIBSVM: A Library for Support Vector Machine*s. 2019.

Hastie, T. *et al*. *The Elements of Statistical Learning*. Springer, 2009.

Hsiang-Fu, Y., H. Fang-Lan y L Chih-Jen. "Dual Coordinate Descent Methods for Logistic Regression and Maximum Entropy Models", en: *Machine Learning*. 85, 2011, pp. 41-75 (DE) https://doi.org/10.1007/s10994-010-5221-8, consultado el 1/08/2020.

Jurafsky, D. y J. H. Martin. *Speech and Language Processing*: An *Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Second Edition. Pearson, 2008.

Measure, A. *Automated Coding of Worker Injury Narratives*. U.S. Bureau of Labor Statistics, 2014.

Platt, J. *Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods*. Advances in Large Margin Classifiers, 1999.

Svensén, M. y C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2007.

Zhang, H. *The Optimality of Naive Bayes*. Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference, FLAIRS 2004. 2, 2004.

Anexo

Para cada proyecto de codificación de encuestas o censos de población y vivienda, se tiene la necesidad de diseñar un conjunto de estrategias y materiales de codificación de acuerdo con las variables principales a codificar, así como sus preguntas de apoyo; para cada uno, se tiene que diseñar un *traje a la medida* para el proceso automático (ver esquema). Para codificar las variables de *ocupación* y *actividad económica* en la ENIGH, bajo el proceso tradicional (algoritmos determinísticos), se utilizan las variables que se muestran en el cuadro 8.

La información se codifica al máximo nivel de desagregación de las clasificaciones utilizadas; para ocupación, se empleó el Sistema Nacional de Clasificación de Ocupación (SINCO) 2011, que tiene 468 grupos unitarios (claves) diferentes y para actividad económica, el Sistema de Clasificación Industrial de América del Norte (SCIAN) 2008 versión hogares, que cuenta con 176 distintos subsectores (ver cuadro 9).

Cuadro 8

Variable	Tipo de variable
Nombre de la ocupación	Texto
Tareas o funciones	Texto
Actividad económica de la empresa o institución	Texto
Nombre de la empresa	Texto
Lugar donde realizó las actividades	Código alfanumérico
Clasificación de la empresa	Código alfanumérico
Trabajo dentro del país	Código alfanumérico
Tamaño de la empresa	Código alfanumérico
Nivel de instrucción	Código alfanumérico
Grados aprobados	Código alfanumérico

Cuadro 9 **Continúa**

Fragmento de una base de datos codificada

var_texto_1	var_texto_2	var_texto_3	var_auxiliar_1	var_auxiliar_2	var_auxiliar_5	Código asignado
auxiliar de almacén	capturar información atender proveedores.	elaboración de carne y productos cárnicos	1	3	1	3132
ayudante de panadería	atender a clientes, acomodar, hacer puestos de pan, preparar el pan relleno, freir donas.	a la elaboración y venta de pan en un local comercial.	1	12	1	4211

Cuadro 9

Concluye

Fragmento de una base de datos codificada

var_texto_1	var_texto_2	var_texto_3	var_auxiliar_1	var_auxiliar_2	var_auxiliar_5	Código asignado
ayudante de albañil	preparar mezcla, pasar y pegar blocs, limpiar la herramienta y preparar cimbra.		2	11	1	9221

Esquema

Proceso de codificación actual para encuestas en hogares del INEGI