ISSN 1870-4069

Técnicas de selección de características y su aplicación en el análisis de polen a través de su textura

Arely Guadalupe Sánchez Méndez, Pedro Arguijo,
José Antonio Hiram Vázquez López, Roberto Ángel Melendez Armenta

Tecnológico Nacional de México, campus Misantla,
División de Ingeniería en Sistemas Computacionales, Misantla, Veracruz,
México

aregpesanmen@gmail.com, pedro_arguijo@excite.com, {jahvazquezl,
ramelendeza}@misantla.tecnm.mx

Resumen. Se propone la implementación de métodos de selección de características para la clasificación de conjuntos de datos de polen basándose en un estudio comparativo entre métodos de selección de características de tipo filtro, envoltura y embebido. Se aplicaron diversos algoritmos de selección de cada método en una base de datos de polen, compuesta por 12 clases de polen que se identifican por 244 características; 34 extraídas mediante el análisis de texturas por medio de la matriz de coocurrencia de niveles de gris (GLCM); y el resto, extraídas mediante patrones locales binarios (LBP). Al aplicar selectores de características en cada conjunto de datos e implementar diferentes clasificadores, se determinó el mejor subconjunto de características de acuerdo a los valores de las métricas de clasificación empleadas. Se obtuvieron resultados competitivos al reducir de 244 a 90 características, aumentando la precisión de clasificación de 75.55% a 77.56%.

Palabras clave: Selección de características, polen, clasificación, GLCM, LBP.

Feature Selection Techniques and their Application in Pollen Analysis through their Texture

Abstract. Feature selection for pollen grain classification is proposed, we consider the selection methods of filter, wrapper, and embedded methods. These selection methods were applied to a dataset containing 12 pollen classes and 244 features per grain; 34 extracted by texture analysis using the gray level co-occurrence matrix (GLCM); and the rest extracted using local binary patterns (LBP). The best subset of features was determined according to the values of the classification metrics used in different classifiers. The results obtained indicate that it is feasible to reduce from 244 to 90 features, increasing the classification accuracy from 75.55% to 77.56%.

Keywords: Feature selection, pollen, classification, GLCM, LBP.

1. Introducción

La palinología es una herramienta importante en el estudio de los granos de polen y otras esporas, permite determinar la familia y género a la cual pertenece la prueba bajo estudio [1]. Entre las aplicaciones de ésta se tiene, por ejemplo, el análisis del polen fósil encontrado en el suelo extraído del fondo de antiguos lagos, el mapeo del clima de miles de años atrás, en el área forense permite determinar la geolocalización de sospechosos o el lugar en el cual se llevó a cabo un crimen, en la agronomía ayuda en la predicción y estudio de contaminantes biológicos.

La clasificación del grano de polen es un proceso cualitativo costoso, que implica la observación e identificación de características por parte de expertos altamente calificados llamados palinólogos. Aunque sigue siendo el método más preciso y eficaz, requiere mucho tiempo y recursos, lo que limita el avance de la investigación.

Los sistemas automáticos que permiten el conteo y clasificación de granos de polen suelen extraer diversas características del polen, ya sean de forma, textura, color o una combinación de estas características [2].

Cuando el número de características aumenta, es factible que algunos atributos sean irrelevantes y/o redundantes, lo cual pueden reducir el desempeño de clasificación. Por lo cual es necesario reducir la dimensionalidad del conjunto de características.

La selección de características es parte del proceso de descubrimiento de conocimientos en conjuntos de datos en el que se selecciona un subconjunto de atributos que son más relevantes para la construcción del modelo predictivo sobre el que se esté trabajando [3].

Existen diversos métodos de selección de atributos, entre los más importantes se encuentran: los de tipo filtro, utilizan un criterio de evaluación basándose únicamente en las propiedades de los datos por lo que son independientes del clasificador; los de tipo envoltura, utilizan el desempeño de algún clasificador para evaluar a los subconjuntos de características; los embebidos que interactúan con el algoritmo de aprendizaje a un costo computacional menor que los de envoltura [4].

En este contexto, el presente trabajo de investigación aborda la selección de características en un conjunto de datos de polen que contiene características concatenadas extraídas por matrices de coocurrencia de niveles de gris (GLCM) y patrones locales binarios (LBP), con la finalidad de encontrar el subconjunto de características que mejore el desempeño de clasificación de los tipos de polen del conjunto de datos que se maneja y, además, identificar las características más relevantes, realizando una comparación entre los resultados de clasificación según el selector de características implementado y el tipo de características seleccionadas.

2. Trabajos relacionados

La investigación se centra en trabajos que apliquen aprendizaje automático en el análisis de conjuntos de datos de polen, tomando como tópico principal la selección de características y clasificación, dando a conocer el estado actual de trabajos realizados relacionados con el tema central de esta investigación.

Tabla 1. Tabla de información del conjunto de datos seleccionado.

	ancho	alto	perimeter	...	energiab	homogeneidadb	clase
0	209	189	770.74935	...	0.003718	0.435142	1
1	207	235	790.991991	...	0.003515	0.434229	1
2	156	211	686.239682	...	0.006662	0.513888	1
3	202	191	713.15137	...	0.004909	0.497538	1
4	222	173	796.991991	...	0.004499	0.461106	1
5	195	178	696.381818	...	0.004001	0.464029	1
6	205	232	901.175757	...	0.004772	0.467111	1
7	268	255	1147.1829	...	0.004171	0.517268	1
8	204	224	846.506709	...	0.004	0.432065	1
9	212	183	719.10974	...	0.005408	0.486413	1

Dell et al. realizaron la discriminación y clasificación de polen utilizando microspectroscopía de infrarrojo por transformada de Fourier de infrarrojo medio junto con métodos estadísticos multivariados supervisados y no supervisados [5].

Mitsumoto et al. desarrollaron un método de clasificación y conteo automático de partículas de polen; por sus resultados, sugieren que un sistema de flujo capaz de medir tanto la luz dispersa como la autoflorescencia de partículas podría clasificar y contar los granos de polen en tiempo real [6].

Chica, M., propone un método basado en el procesamiento de imágenes y la clasificación de una clase para rechazar objetos de granos de polen desconocidos. Aplica algoritmos de selección de características. Como resultado obtuvo una precisión general de clasificación del 92.3% [7].

Hernández y Salamanca, redujeron de 11 variables a cuatro componentes principales mostrando correlación entre variables, además lograron la correcta clasificación del 72.4% de las muestras de polen corbícula en función de los tres factores de preponderancia de los componentes principales y su origen geográfico [8].

Hernández et al. utilizaron el método de PCA y sumaron las salidas de 30 redes neuronales, obteniendo una tasa de éxito del 91.67% ± 3.13% en su clasificación [9].

Popescu y Sasu, automatizaron la clasificación de plantas mediante imágenes de granos de polen. Investigaron la efectividad de 11 algoritmos de extracción/selección de características y de 12 clasificadores basados en aprendizaje automático. Encontraron que algunos de los algoritmos de extracción/selección de características especificados y de clasificación exhibieron un comportamiento consistente [1].

Marcos, et al. realizaron una evaluación exhaustiva sobre la utilidad del análisis de textura para la caracterización automatizada de muestras de polen. Aplicaron cuatro métodos de extracción de características de textura: GLCM, filtros log-Gabor (LGF), LBP y momentos discretos de Tchebichef (DTM). Posteriormente, realizaron reducción de dimensionalidad mediante el análisis discriminante de Fisher y realizaron la clasificación con el algoritmo de K vecinos más cercanos. Sus resultados revelaron que LGF y DTM fueron superiores a GLCM y LBP en la clasificación, descubriendo que la combinación de todas las características de textura dio como resultado el rendimiento más alto, con una precisión del 95% [10].

Del Pozo et al. realizaron preprocesamiento, extracción y clasificación de un conjunto de datos de imágenes de polen. Extrajeron un total de 50 características (24 parámetros geométricos y 26 parámetros de textura extraídos mediante GLCM). Sus

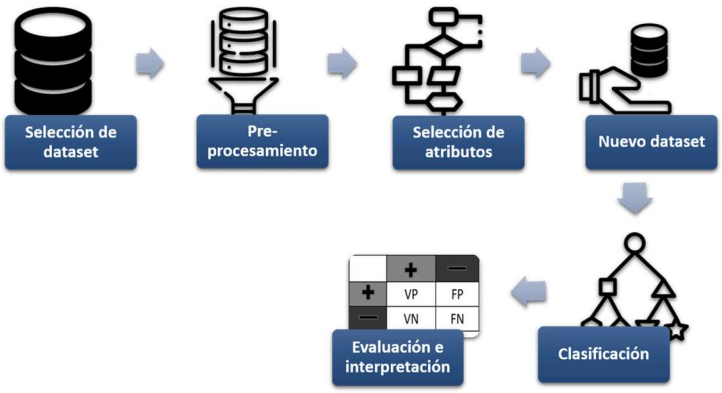

Fig. 1. Diagrama de la metodología implementada.

resultados mostraron tasas de éxito entre 90.54% y 92.81%, destacando la calidad de los parámetros presentados para la clasificación de granos de polen [11].

Goncalves et al. implementaron, ajustaron y probaron una combinación de tres extractores de características: bolsa de palabras visuales (BOW), color, forma y textura (CST) y una combinación de BOW y CST que se llama CST + BOW; cuatro técnicas de aprendizaje automático: dos variaciones de SVM (SMO y C-SVC), un clasificador basado en árbol de decisión (J48) y KNN. Sus resultados mostraron que la tasa de clasificación correcta (CCR) más alta fue del 64% y se logró utilizando CST + BOW y C-SVC [12].

Sevillano y Aznarte, aplicaron tres métodos de clasificación que hacen uso de redes neuronales convolucionales: uno se basa en el aprendizaje por transferencia, otro en extracción de características y un tercero, con un enfoque híbrido, que combina el aprendizaje por transferencia y la extracción de características. Como resultados obtuvieron tasas de clasificación correcta superiores al 97% [13].

3. Materiales y métodos

3.1. Materiales

3.1.1. Selección del conjunto de datos

Se utiliza una base de conocimientos generada como parte del proyecto *Análisis y clasificación de polen*; aprobado por el Tecnológico Nacional de México, el cual consta de 12 clases y cada observación está representada por un vector de 244 características extraídas, de una región de interés de 100×100 pixeles, mediante los métodos de extracción de características de textura: Patrones locales binarios (en inglés Local Binary Patterns, LBP) y Matriz de coocurrencia de escala de grises (en inglés Grey Level Co-ocurrence Matrix, GLCM).

El conjunto de imágenes de las cuales se extrajeron las características de textura indicadas son las reportadas por Tello-Mijares [14]. En total, el conjunto de datos

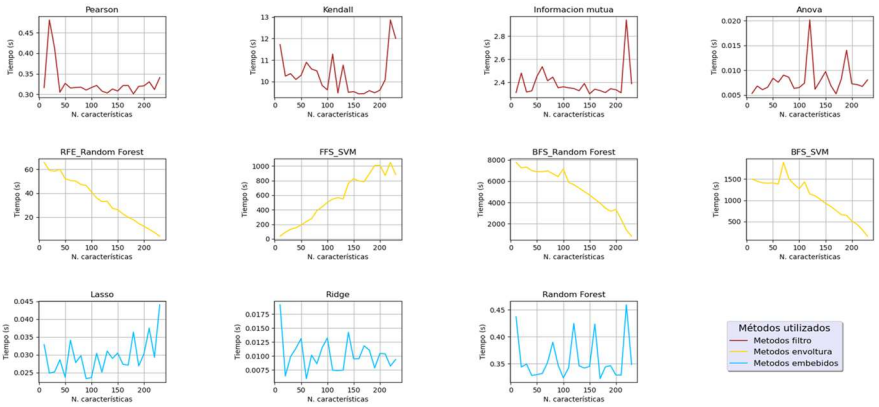

Fig. 2. Tiempos de ejecución de cada selector de características implementado, de acuerdo con el número de características seleccionadas con datos sin normalizar.

utilizado en este trabajo consta de 390 registros, con un promedio de 33 registros por clase. En la Tabla 1 se puede observar una parte del desglose de la información del conjunto de datos, mostrando únicamente 10 registros y seis de las 244 características que lo conforman.

3.1.2. Herramientas

Para el diseño, desarrollo y experimentos del proyecto se utilizaron los siguientes materiales:

- Equipo de cómputo: El análisis de datos, programación y experimentos se realizaron en una computadora con Procesador Intel(R) Core(TM) i7-7500U CPU 2.70 GHz, Microsoft Windows 10 Home Single Language 64-bits, RAM 16GB DDR4 2400 MHz, SSD 128 GB y HDD 1 TB.
- Software: Para el manejo de archivos que contienen a los conjuntos de datos se utiliza Excel. Para la programación y experimentos se utiliza Spyder con Python 3.7.6.
- Librerías de Python:
  - scikit-learn: Cuenta con algoritmos de clasificación, regresión, agrupamiento y reducción de dimensionalidad. Además, es compatible con otras librerías de Python como NumPy, SciPy y matplotlib.
  - Mlxtend: Implementa una variedad de algoritmos y utilidades centrales para máquinas aprendizaje y minería de datos.

3.2. Métodos

La metodología propuesta para el presente trabajo consta de las fases del proceso de descubrimiento de conocimientos en bases de datos, representadas en el esquema de la Fig. 1.

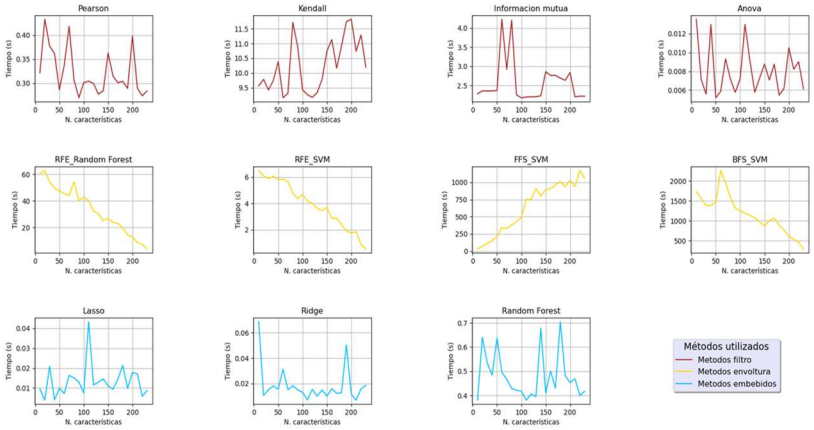

Fig. 3. Tiempos de ejecución de cada selector de características implementado, de acuerdo con el número de características seleccionadas con datos normalizados.

3.2.1 Preprocesamiento

Con el fin de asegurar la utilidad de los datos, se realiza la limpieza de estos. Para experimentos de la investigación se normalizan los datos como técnica de transformación, ya que se realizará una comparación entre resultados utilizando datos sin normalizar y normalizados.

El resultado de esta etapa son dos archivos con extensión .CSV que contienen los datos adecuados para la etapa de selección de atributos relevantes.

3.2.2 Conjuntos de entrenamiento y prueba

Para determinar qué elementos pertenecen a cada uno de los conjuntos se estableció que para el conjunto de entrenamiento se utilice 70% del conjunto original y el 30% restante representa el conjunto de prueba.

3.2.3 Selección de características

Se aplican los algoritmos de selección de características, elegidos de acuerdo con el análisis bibliográfico realizado [15, 16], los cuales se enlistan a continuación.

- De tipo filtro: Coeficiente de correlación (Pearson y Kendall), información mutua y análisis de varianza.
- De tipo envoltura: RFE con RF, FFS con RF, FFS con SVM, BFS con RF y BFS con SVM.
- De tipo embebido: LASSO, Ridge y RF.

Con esto, los experimentos se realizan con los algoritmos de selección de atributos aplicados a cada conjunto de datos.

Tabla 2. Valores de las medidas de clasificación para el conjunto de datos con 90 características seleccionadas del conjunto con datos sin normalizar, de acuerdo al algoritmo de selección implementado.

		SELECTORES DE CARACTERÍSTICAS										
		FILTRO				WRAPPER				EMBEDDED		
		Correlación		IM	Anova	RFE	FFS	BFS		Lasso	Ridge	RF
		Pearson	Kendall			RF	SVM	RF	SVM			
Tiempo seleccionador		0.3104	9.8201	2.3520	0.0063	46.4293	439.0289	6431.0504	1372.8778	0.0233	0.0115	0.3478
CLASIFICADOR	MEDIDAS											
Random Forest	Exactitud	70.94%	75.21%	70.09%	69.23%	75.21%	70.94%	71.80%	70.94%	64.96%	71.80%	76.07%
	Precisión	70.13%	77.24%	74.94%	70.81%	75.37%	73.33%	71.69%	73.63%	63.64%	73.95%	79.22%
	Exhaustividad	67.53%	75.05%	73.19%	72.40%	77.24%	74.35%	74.28%	73.87%	67.97%	75.03%	79.26%
	Tiempo (s)	0.027	0.022	0.024	0.024	0.026	0.030	0.023	0.022	0.022	0.033	0.026
Adaboost SVM	Exactitud	64.53%	68.80%	63.68%	67.95%	64.53%	67.09%	64.53%	67.09%	64.53%	72.22%	64.53%
	Precisión	54.63%	56.25%	53.83%	55.45%	54.63%	53.49%	54.63%	53.49%	54.63%	55.19%	54.63%
	Exhaustividad	58.41%	61.61%	57.76%	61.33%	58.41%	60.86%	58.41%	60.86%	58.41%	63.82%	58.41%
	Tiempo (s)	2170.615	2821.619	2195.823	2202.106	2376.360	2378.517	2611.610	2840.510	2378.508	2384.821	2210.515
Adaboost Perceptron	Exactitud	41.97%	58.21%	64.19%	77.01%	70.17%	52.22%	60.77%	60.77%	57.35%	74.44%	62.48%
	Precisión	31.82%	50.92%	63.78%	83.15%	75.28%	66.80%	60.84%	60.84%	67.64%	76.97%	61.62%
	Exhaustividad	38.80%	53.78%	64.36%	77.72%	69.55%	55.48%	57.81%	57.81%	58.98%	73.85%	63.44%
	Tiempo (s)	11.662	10.486	20.972	22.442	22.344	24.402	18.718	24.990	19.502	22.736	24.304
SVM	Exactitud	44.80%	47.37%	42.24%	61.04%	75.57%	42.24%	44.80%	42.24%	44.80%	49.08%	44.80%
	Precisión	35.22%	45.49%	33.87%	50.37%	78.45%	31.43%	35.22%	31.43%	35.22%	31.89%	35.22%
	Exhaustividad	45.94%	49.08%	39.46%	58.32%	78.52%	42.24%	45.94%	42.24%	45.94%	46.48%	45.94%
	Tiempo (s)	1033.626	1343.628	1045.630	1048.622	1131.600	1132.627	1243.624	1352.624	1132.623	1135.629	1052.626
KNN	Exactitud	65.90%	65.90%	51.37%	60.77%	65.90%	50.51%	65.04%	50.51%	65.04%	64.19%	65.90%
	Precisión	62.96%	60.27%	50.66%	51.13%	62.96%	45.21%	64.31%	45.21%	64.31%	63.16%	62.96%
	Exhaustividad	68.97%	65.33%	52.80%	57.66%	68.97%	53.09%	66.13%	53.09%	66.13%	64.08%	68.97%
	Tiempo (s)	2.046	2.232	2.232	2.418	2.790	2.232	2.418	2.232	2.046	2.418	2.604
Extra Tree	Exactitud	69.23%	69.23%	68.38%	68.38%	**77.78%**	69.23%	70.94%	69.23%	60.68%	74.36%	74.36%
	Precisión	68.29%	72.39%	68.05%	70.09%	**77.56%**	70.69%	72.52%	69.03%	58.51%	73.87%	74.76%
	Exhaustividad	68.87%	66.34%	68.10%	71.47%	**79.79%**	69.77%	74.40%	72.77%	60.87%	78.22%	76.61%
	Tiempo (s)	1.514	1.422	1.439	1.506	**1.642**	1.680	1.437	1.664	1.417	1.754	1.637
Multi-Layer Perceptron	Exactitud	23.93%	29.06%	35.95%	22.22%	66.46%	32.48%	47.66%	36.55%	33.33%	63.25%	29.92%
	Precisión	15.18%	26.26%	39.02%	20.42%	68.18%	22.06%	37.74%	31.58%	26.40%	62.07%	25.54%
	Exhaustividad	20.08%	25.78%	42.59%	22.97%	61.07%	27.81%	43.31%	34.09%	26.38%	64.15%	27.86%
	Tiempo (s)	0.096	0.373	0.227	0.216	0.318	0.237	0.061	0.069	0.153	3.012	0.137
Naive Bayes	Exactitud	38.46%	45.30%	38.46%	52.99%	69.03%	41.03%	39.32%	41.03%	39.32%	70.09%	38.46%
	Precisión	33.98%	40.33%	31.06%	46.41%	63.40%	35.87%	33.58%	35.87%	33.71%	67.29%	33.41%
	Exhaustividad	36.91%	42.56%	34.11%	51.71%	65.64%	38.93%	36.57%	38.93%	36.62%	72.11%	34.44%
	Tiempo (s)	0.008	0.008	0.008	0.008	0.007	0.010	0.008	0.009	0.009	0.009	0.013

3.2.4 Nuevo conjunto de datos

Se utilizan diferentes números de características seleccionadas por cada algoritmo, de manera que se generarán nuevos conjuntos de datos, de acuerdo con el algoritmo y número de atributos seleccionados.

Cada conjunto de datos nuevo debe contener los atributos mejor rankeados y/o seleccionados que resulten de la implementación de los algoritmos de selección de atributos. Por ejemplo, para el conjunto de datos empleado, aplicando el algoritmo de información mutua (método de selección de características), se seleccionan 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20 y 21 características en diferentes ejecuciones del algoritmo, por lo que se obtendrán 12 nuevos conjuntos de datos, cada uno con las características seleccionadas según la cantidad que se indicó antes de ejecutar el algoritmo de selección de características.

3.2.5 Clasificación

Se implementa un modelo de clasificación aplicando los algoritmos que a continuación se enlistan.

Tabla 3. Valores de las medidas de clasificación para el conjunto de datos con 90 características seleccionadas del conjunto con datos normalizados, de acuerdo con el algoritmo de selección implementado.

		SELECTORES DE CARACTERÍSTICAS										
		FILTRO				WRAPPER				EMBEDDED		
		Correlación		IM	Anova	RFE	FFS	BFS		Lasso	Ridge	RF
		Pearson	Kendall			RF	SVM	RF	SVM			
Tiempo seleccionador		0.2697	10.8965	2.2500	0.0058	40.1889	4.3601	427.8767	1318.8640	0.0130	0.0149	0.4211
CLASIFICADOR	MEDIDAS											
Random Forest	Exactitud	60.43%	61.28%	45.90%	55.30%	57.86%	57.01%	55.30%	55.30%	54.44%	57.86%	57.86%
	Precisión	58.10%	58.59%	44.69%	55.50%	57.31%	55.60%	56.99%	56.07%	53.47%	57.02%	56.45%
	Exhaustividad	60.40%	61.56%	48.79%	58.23%	60.16%	58.74%	57.77%	56.77%	53.87%	59.93%	59.93%
	Tiempo (s)	1.739	1.696	1.770	2.089	2.260	2.379	2.904	1.915	3.267	3.436	3.632
Adaboost SVM	Exactitud	63.68%	64.53%	62.82%	62.82%	63.68%	63.68%	61.11%	57.69%	62.82%	63.68%	63.68%
	Precisión	53.19%	55.28%	52.00%	52.06%	53.19%	53.19%	50.96%	51.99%	52.06%	53.19%	53.19%
	Exhaustividad	60.18%	61.57%	60.71%	59.26%	60.18%	60.18%	58.33%	61.61%	59.26%	60.18%	60.18%
	Tiempo (s)	1941.884	1946.567	1935.640	2230.669	2374.281	2240.035	2127.643	1996.519	3101.707	4110.113	3246.880
Adaboost Perceptron	Exactitud	41.37%	38.80%	31.11%	34.53%	48.21%	43.08%	31.97%	32.82%	37.09%	40.51%	39.66%
	Precisión	28.98%	30.15%	20.98%	28.31%	47.75%	39.86%	25.35%	28.79%	24.46%	40.19%	28.19%
	Exhaustividad	38.64%	43.26%	28.33%	37.21%	43.12%	41.36%	32.29%	32.56%	35.90%	38.44%	36.09%
	Tiempo (s)	8.585	6.715	2.040	9.690	29.240	62.050	12.240	2.890	4.675	17.765	22.950
SVM	Exactitud	64.79%	62.22%	51.11%	63.08%	64.79%	64.79%	51.11%	51.11%	63.08%	64.79%	64.79%
	Precisión	54.82%	52.64%	41.01%	48.28%	54.82%	54.82%	40.96%	40.96%	48.28%	54.82%	54.82%
	Exhaustividad	60.42%	59.35%	48.33%	58.49%	60.42%	60.42%	48.33%	48.33%	58.49%	60.42%	60.42%
	Tiempo (s)	32.032	29.120	34.944	37.856	59.696	40.768	36.400	33.488	43.680	52.416	45.136
KNN	Exactitud	53.46%	50.04%	38.93%	48.33%	53.46%	53.46%	51.75%	52.61%	48.33%	53.46%	53.46%
	Precisión	52.75%	49.41%	34.16%	51.87%	56.62%	56.62%	48.49%	47.90%	52.16%	56.62%	56.62%
	Exhaustividad	48.11%	44.28%	36.01%	45.72%	48.17%	48.17%	49.62%	51.86%	45.72%	48.17%	48.17%
	Tiempo (s)	0.938	0.871	1.072	0.871	1.072	1.206	0.871	0.938	1.206	1.206	1.273
Extra Tree	Exactitud	**69.83%**	68.97%	47.61%	57.86%	57.86%	54.44%	55.30%	56.15%	58.72%	53.59%	54.44%
	Precisión	**70.06%**	70.06%	48.19%	59.62%	58.95%	55.08%	57.47%	57.45%	57.09%	52.57%	53.78%
	Exhaustividad	**71.83%**	72.52%	47.58%	59.03%	58.17%	56.02%	57.08%	58.13%	56.94%	55.38%	55.41%
	Tiempo (s)	**1.362**	1.403	1.711	1.833	2.520	1.875	1.570	1.447	2.270	2.114	2.561
Multi-Layer Perceptron	Exactitud	42.22%	43.93%	35.38%	43.08%	43.08%	43.08%	33.68%	35.38%	43.08%	42.22%	44.79%
	Precisión	28.08%	30.51%	21.28%	28.88%	28.81%	30.10%	22.66%	23.47%	31.21%	28.57%	29.75%
	Exhaustividad	37.95%	40.05%	28.33%	38.79%	38.92%	38.73%	31.40%	30.47%	38.79%	38.28%	40.21%
	Tiempo (s)	0.528	0.478	0.177	0.620	0.620	0.733	0.562	0.224	0.858	0.789	0.996
Naive Bayes	Exactitud	60.88%	62.59%	49.77%	50.62%	56.61%	56.61%	55.75%	49.77%	55.75%	56.61%	56.61%
	Precisión	53.92%	56.43%	43.61%	40.75%	46.85%	46.85%	41.96%	39.20%	46.11%	46.85%	46.85%
	Exhaustividad	61.04%	63.21%	49.41%	51.53%	56.37%	56.37%	55.92%	49.33%	55.99%	56.37%	56.37%
	Tiempo (s)	0.006	0.008	0.009	0.006	0.008	0.012	0.010	0.008	0.016	0.007	0.014

- RF.
- AdaBoost.
- MLP.
- NB.
- KNN.
- Extra trees classifier.
- SVM.

3.2.6 Evaluación

Se analizan los resultados que se obtuvieron utilizando las medidas de precisión de los algoritmos, ofreciendo argumentos y valoraciones que demuestren la factibilidad y precisión de la clasificación.

4. Experimentos y resultados

4.1. Selección de características

Se utiliza el conjunto de datos sin ser normalizados y con datos normalizados. Al implementar cada uno de los algoritmos seleccionados para realizar la selección de

variables, uno de los argumentos de entrada en el que todos coinciden es el número de características a ser seleccionadas.

Para poder identificar un número óptimo de características seleccionadas, se han generado conjuntos de datos de 10 hasta 230 características, considerando que el conjunto de datos está conformado por 244 características. Posteriormente se realiza la clasificación de cada uno de los conjuntos de datos generados y la comparativa de los valores métricos para cada modelo.

De acuerdo con el tiempo de ejecución de los diferentes algoritmos de selección de características implementados en el conjunto de datos sin normalizar, se presentan las gráficas de la Fig. 2 donde se muestra el tiempo de ejecución (en segundos) en función de la cantidad de características seleccionadas (de 10,20,30,40, ..., 230). Las gráficas se analizan de acuerdo con el tipo de algoritmo: de color marrón los de tipo filtro, de amarillo los de envoltura y de azul los embebidos.

Los tiempos de ejecución para los métodos filtro van de 0.0052 a 12.8686 segundos; los de envoltura van de 4.0287 a 7774.5830 segundos; los de los embebidos su mayor tiempo de ejecución fue de 0.4594 y el menor fue de 0.0059. De los datos anteriores, de manera general, se puede identificar que los algoritmos con mayor tiempo de ejecución son los de tipo envoltura.

El algoritmo de selección de características que requirió más tiempo de ejecución, con 7774.583 segundos fue el de BFS implementando RF con 10 características. El algoritmo de selección de características que empleó menos tiempo de ejecución, con 0.0052 segundos, fue el de ANOVA con 170 características seleccionadas.

En la Fig. 3, el gráfico representa los tiempos de ejecución de los algoritmos de selección de características implementados en el conjunto de datos normalizado. Los tiempos de ejecución para los métodos filtro van de 0.0051 a 11.8254 segundos; los de envoltura van de 0.5302 a 2266.8091 segundos; los de los embebidos su mayor tiempo de ejecución fue 0.7050 y el menor fue de 0.0038 segundos.

De los datos anteriores, de manera general, se puede identificar que los algoritmos con mayor tiempo de ejecución son los de tipo envoltura. El algoritmo de selección de características que requirió más tiempo de ejecución, con 2266.8091 segundos fue el de BFS implementando SVM con 60 características. El algoritmo de selección de características que utilizó menos tiempo de ejecución, con 0.0038 segundos, fue el de LASSO con 20 características seleccionadas (con BFS-SVM seleccionando 20 características le llevó un tiempo de ejecución de 1560.9076 segundos).

Los resultados de tiempo de ejecución serán mejor valorados al momento de comparar los resultados de las medidas obtenidas en la clasificación de los datos considerando el número de características seleccionadas e implementando diversos algoritmos de clasificación.

4.2. Clasificación

Los resultados correspondientes a la clasificación de los datos sin normalizar y sin realizar la selección de características. El mayor porcentaje de exactitud en la clasificación de los datos ha sido del 75.21% obtenido por el clasificador RF.

Al implementar cada algoritmo de selección de características elegido, se crearon 23 nuevos conjuntos de datos, de acuerdo con el número de características seleccionadas

(desde 10 hasta 230 características). Seguido de esto, se realizó la clasificación de los datos, utilizando los algoritmos de clasificación seleccionados.

De cada nuevo conjunto de datos creado según el número de características seleccionadas con datos no normalizados, se procedió a realizar la clasificación de estos, aplicando los ocho clasificadores seleccionados para este fin. Se eligió al clasificador que diera mejores resultados en sus medidas de clasificación definidas, incluso el tiempo de ejecución al realizar la selección de características.

Al realizar la clasificación de los datos, el mayor porcentaje en exactitud, precisión y exhaustividad con 77.78%, 77.56% y 79.79%, respectivamente, se obtuvo cuando se seleccionaron 90 características.

En la Tabla 2 se presentan los valores de las medidas de clasificación para el conjunto de datos con 90 características seleccionadas del de datos no normalizados; el clasificador con mejor desempeño fue Extra Tree Classifier cuando se realizó la selección de características mediante el método de envoltura RFE implementado con Random Forest, el clasificador tuvo un tiempo de ejecución de 1.64 segundos y el del seleccionador de características fue de 46.4293 segundos (un tiempo de ejecución menor a un minuto, lo cual no es tan lento en comparación con los otros métodos de envoltura que llegaron a tardar más de dos horas).

Cuando se realizó la clasificación del conjunto de datos normalizado y sin realizar la selección de características. El mayor porcentaje de exactitud en la clasificación de los datos ha sido del 75.72% obtenido por el clasificador Extra Tree Classifier.

Al implementar los algoritmos de clasificación a los 23 nuevos conjuntos con datos normalizados, después de realizar la selección de características, se eligió al clasificador que diera mejores resultados en sus medidas de clasificación definidas, incluso el tiempo de ejecución al realizar la selección de características. Al realizar la clasificación de los datos, el mayor porcentaje en exactitud, precisión y exhaustividad con 69.83%, 70.06% y 71.83%, respectivamente, se obtuvo cuando se seleccionaron 90 características.

En la Tabla 3 se presentan los valores de las medidas de clasificación para el conjunto de datos con 90 características seleccionadas del conjunto con datos normalizados. El clasificador con mejor desempeño fue Extra Tree Classifier cuando se realizó la selección de características mediante el método filtro de coeficiente de correlación de Pearson, el clasificador tuvo un tiempo de ejecución de 1.362 segundos y el del seleccionador de características fue de 0.2697 segundos.

5. Conclusiones y trabajo a futuro

Se presenta la implementación de tres métodos de selección de características (filtro, envoltura y embebido) aplicando diferentes algoritmos de cada tipo. Con el objetivo de demostrar que para un conjunto completo de características iniciales existe un subconjunto de características que mejore o mantenga el desempeño de clasificación de los tipos de polen que se manejan en el conjunto de datos con características extraídas mediante el GLCM y LBP.

Se aplicaron cuatro algoritmos de selección de características de tipo filtro, tres de tipo envoltura y tres de tipo embebido. La mejor clasificación con los 244 atributos se obtuvo con el clasificador Random Forest con 75.21%, 75.55% y 77.45% de exactitud,

precisión y exhaustividad, respectivamente. Cuando se aplicaron los selectores de características, la mejor clasificación se obtuvo cuando se manejó un subconjunto de 90 características extraídas mediante el algoritmo RFE con Random Forest de tipo envoltura y se utilizó el clasificador Extra Tree; los valores de clasificación fueron de 77.78% en exactitud, 77.56% en precisión y 79.79% en exhaustividad.

Se puede concluir aquí que la selección de características mejoró los valores de las métricas de clasificación implementadas. Es importante mencionar que se obtuvieron valores muy cercanos e incluso mayores a la clasificación obtenida al aplicar la selección de características en este conjunto de datos cuando se seleccionaron las características mediante el algoritmo de correlación con el coeficiente de Kendall (método filtro) y los algoritmos Ridge y Random Forest que son de tipo embebido, además de que también funcionó como buen clasificador el Random Forest.

Finalmente se puede decir que el resultado de clasificación mejora si se utilizan métodos de selección que proporcionen buenos subconjuntos de características. Los resultados de los experimentos muestran que los algoritmos de tipo filtro, embebido y de envoltura funcionan bien con este tipo de datos, aunque los de tipo embebido puedan mejorar un poco los porcentajes de correcta clasificación, tienen el punto en contra de que llevan más tiempo de ejecución.

Por ejemplo, cuando se seleccionaron 90 características con valores de clasificación similares, el algoritmo RFE-RF (de tipo envoltura) le tomó 46.42 segundos, mientras que con el algoritmo de correlación con el coeficiente Kendall (de tipo filtro) fue de 9.82 segundos y con el de Random Forest(embebido) le tomó 0.34 segundos.

Con la finalidad de analizar la posibilidad de obtener mejores resultados de clasificación se realizó la normalización de los datos, pero al comparar los resultados de clasificación con los datos sin normalizar, se concluye que, el realizar la normalización de los datos disminuyó el porcentaje de clasificación.

Como trabajo futuro se propone realizar un sistema automatizado que permita el conteo de polen implementando el proceso de selección de características y considerando cuáles selectores y clasificadores funcionaron mejor en este trabajo de investigación, con la finalidad de mejorar la clasificación y correcta identificación de polen.

Referencias

1. Popescu, M. C., Sasu, L. M.: Feature extraction, feature selection and machine learning for image classification: A case study. In: Proceedings of International Conference on Optimization of Electrical and Electronic Equipment (OPTIM), IEEE, pp. 968–973 (2014) doi: 10.1109/OPTIM.2014.6850925
2. Damián, M. R.: Clasificación automática y recuento de granos de polen a partir de imágenes digitales de microscopía óptica. Tesis Doctoral, Universidade de Vigo (2005)
3. Kotsiantis, S.: Feature selection for machine learning classification problems: A recent overview. Artificial Intelligence Review, vol. 42, no 1, pp. 157–176 (2011) doi: 10.1007/s10462-011-9230-1
4. Hall, M. A.: Correlation-based feature selection for discrete and numeric class machine learning. In: Proceedings of the Seventeenth International Conference on Machine Learning, pp. 359–366 (2000)
5. Dell'Anna, R., Lazzeri, P., Frisanco, M., Monti, F., Campeggi, F. M., Gottardini, E., Bersani, M.: Pollen discrimination and classification by Fourier transform infrared (FT-IR)

microspectroscopy and machine learning. Analytical and bioanalytical chemistry, vol. 394, pp. 1443–1452 (2009) doi: 10.1007/s00216-009-2794-9
6. Mitsumoto, K., Yabusaki, K., Aoyagi, H.: Classification of pollen species using autofluorescence image analysis. Journal of bioscience and bioengineering, vol. 107, no 1, pp. 90–94 (2009) doi: 10.1016/j.jbiosc.2008.10.001
7. Chica, M.: Authentication of bee pollen grains in bright-field microscopy by combining one-class classification techniques and image processing. Microscopy research and technique, vol. 75, no. 11, pp. 1475–1485 (2012) doi: 10.1002/jemt.22091
8. Hernández, J., Salamanca, G.: Métodos estadísticos para la clasificación fisicoquímica de polen corbicular de la zona altoandina de Boyacá, Colombia (2012)
9. Hernández, J. A., González, C. M. T., Rivas, J. T., Mora, F. M., Huertas, O. S., Bogantes, M. R., Chavez, L. S.: Sistema de detección y clasificación automática de granos de polen mediante técnicas de procesado digital de imágenes. Uniciencia, vol. 27, no. 1, pp.59–73 (2013)
10. Marcos, J. V., Nava, R., Cristóbal, G., Redondo, R., Escalante-Ramírez, B., Bueno, G., Déniz, O., González-Porto, A., Pardo, C., Chung, F., Rodriguez, T.: Automated pollen identification using microscopic imaging and texture analysis. Micron, vol. 68, pp. 36–46 (2015) doi: 10.1016/j.micron.2014.09.002
11. del Pozo-Banos, M., Ticay-Rivas, J. R., Alonso, J. B., Travieso, C. M.: Features extraction techniques for pollen grain classification. Neurocomputing, vol. 150, pp. 377–391 (2015) doi: 10.1016/j.neucom.2014.05.085
12. Barbosa, A., Silva, J., Goncalves, G., Pascoli, M., Pott, A., Hirosi, M., Pistori, H.: Feature extraction and machine learning for the classification of Brazilian savannah pollen grains. PlOS ONE, vol. 11, no. 6, pp. 1–20 (2016) doi: 10.1371/journal.pone.0157044
13. Sevillano, V., Aznarte, J. L.: Improving classification of pollen grain images of the POLEN23E dataset through three different applications of deep learning convolutional neural networks. PLOS ONE, vol. 13, no. 9 (2018) doi: 10.1371/journal.pone.0201807
14. Tello-Mijares, S., Flores, F.: A novel method for the separation of overlapping pollen species for automated detection and classification. Computational and mathematical methods in medicine (2016) doi: 10.1155/2016/5689346
15. Bolón-Canedo, V., Sánchez-Maroño, N., Alonso-Betanzos, A.: Feature selection for high-dimensional data. Cham, Springer International Publishing (2015)
16. Bolón-Canedo, V., Alonso-Betanzos, A.: Recent advances in ensembles for feature selection. Berlin, Heidelberg, Springer, vol. 147 (2018)