DETECTOR PREDICTIVO DE CONEXIONES FRAUDULENTAS

Alumno: David Martin Tinaquero

Máster en Seguridad de las TIC

Aplicación de técnicas de *Mahine Learning* a la Seguridad

Director de TFM: Enric Hernández

Tutor de la asignatura: Víctor García Font

GUIÓN

- Presentación de los objetivos
- Datos de entrenamiento
  - Características
  - Obtención
  - Asignación de los datos a las fases
- Estructuración de datos
  - BD NoSQL Cassandra
- Clasificador predictivo
  - Tipos de algoritmos ML (*Machine Learning*)
  - Tipos de algoritmos supervisados
  - *Deep Learning*
  - Lenguajes de programación
  - Librerías
  - Diseño e implementación del modelo
  - Métricas de evaluación del modelo final
  - Reutilización del modelo
  - Diagrama del modelado y uso
- Conclusiones

OBJETIVOS

- Diseño e implementación de un detector predictivo de conexiones de red fraudulentas.
- Características del producto a desarrollar:
  - Porcentaje de acierto elevado.
  - Porcentaje de falsos positivos mínimo o nulo.
  - Autónomo. Capacidad de aprendizaje para detectar automáticamente nuevos patrones.
  - Eficiente. Clasificación de grandes cantidades de tramas en tiempo reducido.
  - Escalable. Posibilidad de ampliación.

DATOS DE ENTRENAMIENTO
CARACTERÍSTICAS

- Los datos para entrenar al modelo clasificador predictivo tiene que cumplir con:
  - **Cantidad.** Suficiente y variada.
  - **Calidad.** Muestra representativa fiel a la realidad que se modela.
- La naturaleza del dato puede ser:
  - **Real.** Ej. Capturas de logs.
  - **Sintética.** Generados mediante procesos explícitos para tal fin.
- Es importante saber de los datos que:
  - **Nunca** contendrán todos los posibles vectores de características de entrada.

DATOS DE ENTRENAMIENTO
OBTENCIÓN

Los datos utilizados en el proyecto fueron proporcionados por el Programa de Evaluación de Detección de intrusos DARPA para el concurso de detección de intrusos KDD de 1999. Link: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

El juego consta de los siguientes ficheros:

- kddcup.data.~5MM de registros de 42 campos. Etiquetados con 22 tipos de ataque.
- kddcup.data_10_percent. ~10% del fichero anterior.
- kddcup.newtestdata_10_percent_unlabeled. ~10% del fichero siguiente.
- kddcup.testdata_unlabeled. ~3MM de registros sin etiquetar. 14 tipos de ataque nuevos, no presentes en el fichero kddcup.data.
- kddcup.testdata_10_percent_unlabeled. ~10% del fichero anterior.
- corrected. ~10% del fichero kddcup.testddata_unlabeled con etiquetas.

DATOS DE ENTRENAMIENTO
ASIGNACIÓN DE LOS DATOS A LAS FASES

Fases y ficheros seleccionados para cada una:

- **Entrenamiento**: kddcup.data_10_percent
- **Validación**: subconjunto aleatorio del 20% de kddcup.data_10_percent
- **Test**: corrected
- **Predicción**: kddcup.newtestdata_10_percent_unlabeled

ESTRUCTURACIÓN DE LOS DATOS
BD NOSQL CASSANDRA

Características:

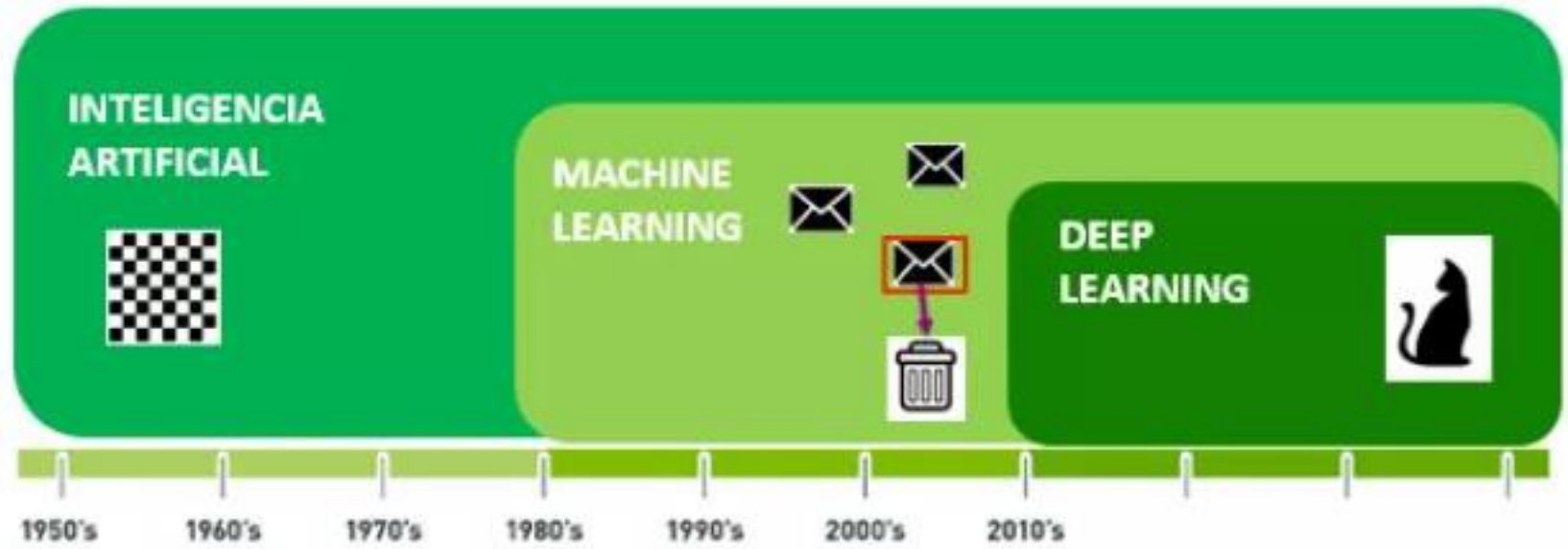
- **Alta disponibilidad.** Debido a su naturaleza distribuida.
- **Tolerancia a fallos.** Mediante replicación automática en otros nodos.
- **Rendimiento.** Capaz de gestionar grandes volúmenes de datos.
- **Descentralizada.** Evita fallos masivos.
- **Escalabilidad.** > 2.000 nodos, > 400 TB de dados y ~1 billón de solicitudes/día
- **Lenguaje CQL**. Muy parecido al SQL. Reducida curva de aprendizaje.

CLASIFICADOR PREDICTIVO
TIPOS DE ALGORITMOS ML

CLASIFICADOR PREDICTIVO
TIPOS DE ALGORITMOS SUPERVISADOS

- Regresión logística
- Basados en instancia
- Árboles de decisión
- Bayesianos
- SVM (*Suport Vector Machine*)
- Redes neuronales
- *Deep Learning*

CLASIFICADOR PREDICTIVO
DEEP LEARNING



CLASIFICADOR PREDICTIVO
LENGUAJES DE PROGRAMACIÓN

CLASIFICADOR PREDICTIVO
LENGUAJES DE PROGRAMACIÓN

Lenguaje	Velocidad	Curva de aprendizaje	Preparado para producción	Soporte de la comunidad	Coste económico	Soporte frameworks DNN	Total
Python	0	1	1	1	1	1	5
R	1	1	0	1	1	0	4
Octave	1	1	0	0	1	0	3
Matlab	1	1	0	1	0	0	3

CLASIFICADOR PREDICTIVO
LIBRERÍAS

CLASIFICADOR PREDICTIVO
LIBRERÍAS

TENSORFLOW

- Fácil implementación.
- API de alto nivel para usar y compartir.
- Gestión del ciclo de vida para desarrolladores.
- Soporte para GPU.
- Mejor soporte para configuraciones de máquinas múltiples.

CLASIFICADOR PREDICTIVO
DISEÑO E IMPLEMENTACIÓN DEL MODELO

FASES

- Importación de datos
- Asignación de valores iniciales a los hiperparámetros
- Transformación de variables categóricas
- Entrenamiento
- Evaluación
- Test
- Ajuste y selección del modelo más optimo

CLASIFICADOR PREDICTIVO
MÉTRICAS DE EVALUACIÓN DEL MODELO FINAL

MATRIZ DE CONFUSIÓN

		Predicción		
		Positivos Pred.	Negativos Pred.	
Observación	Positivos Obs.	Verdaderos Positivos (VP) = 227.632	Falsos Negativos (FN) (Error Tipo II) = 22.804	Valor Predictivo Positivo = VP / Positivos Obs. = **90,89%**
	Negativos Obs.	Falsos Positivos (FP) (Error Tipo I) = 536	Verdaderos Negativos (VN) = 60.057	Valor Predictivo Negativo = VN / Negativos Obs. = **99,12%**
		Sensibilidad = VP / Positivos Pred. = **99,77%**	**Especificidad =** VN / Negativos Pred. = **72,48%**	

CLASIFICADOR PREDICTIVO
MÉTRICAS DE EVALUACIÓN DEL MODELO FINAL

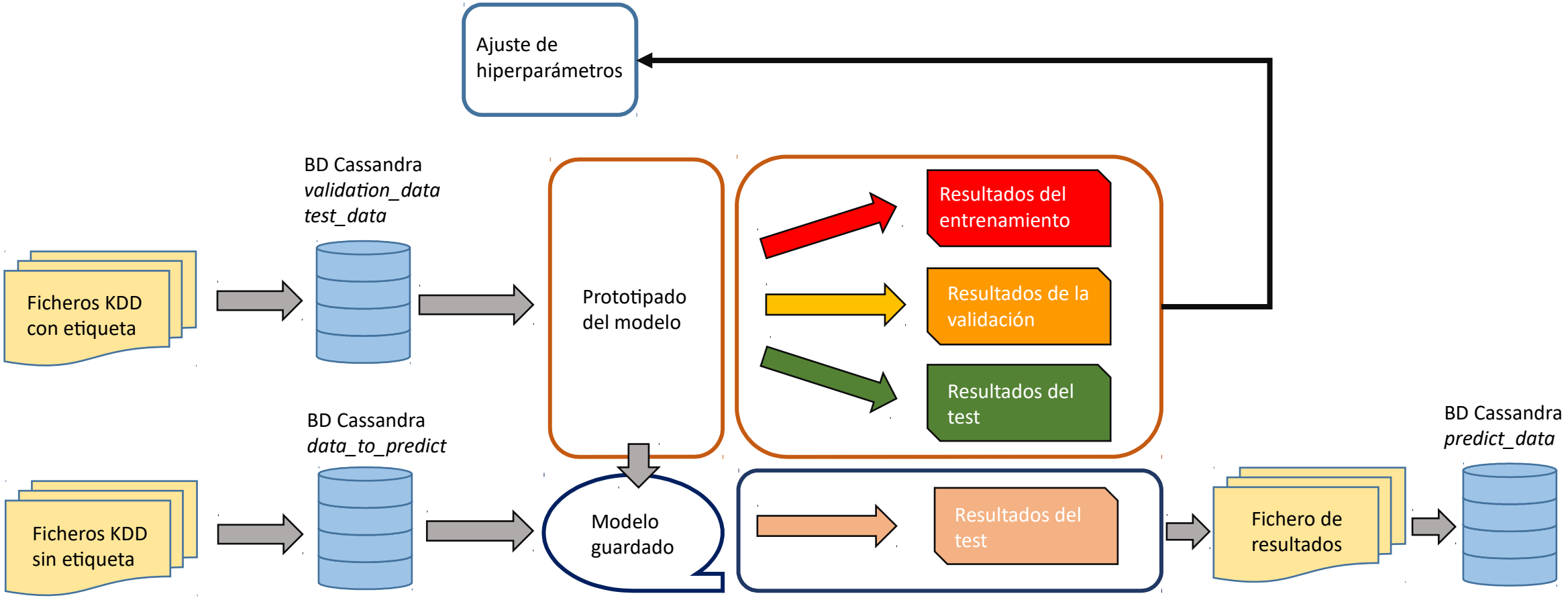
RESULTADOS DE EJECUCIÓN

```
***** TIEMPOS *****
Inicio del proceso: 02/06/2018 16:58:48
Tiempo empleado en la importación de datos de entrenamiento: 0:00:01.528826
Tiempo empleado en la importación de datos de validación: 0:00:00.078975
Tiempo empleado en la importación de datos de test: 0:00:01.245614
Tiempo empleado en la fase de entrenamiento: 0:04:33.822855
Tiempo empleado en la fase de validación: 0:00:07.887856
Tiempo empleado en la fase de predicción: 0:00:21.262381
***** PRECISIÓN *****
VN= 60057, FP = 536, FN = 22804, VP = 227632
Valor de predicción positivo: 90.89428037502596
Valor de predicción negativo: 99.11540937071939
Sensibilidad: 99.76508537568809
Especificidad: 72.47921217460566
Error Tipo I: 536
Error Tipo II: 22804
Precisión en la validación: 0.9926318526268005
Precisión en el test: 0.9249587655067444
Fin del proceso: 02/06/2018 17:04:19
Tiempo total empleado en la ejecución: 0:05:30.885228
***** FIN *****
```

CLASIFICADOR PREDICTIVO
REUTILIZACIÓN DEL MODELO

- Clasificación de ~300.000 tramas de red
- Fases:
  - Importación de fichero de datos a BD
  - Predicción con el modelo clasificador
  - Exportación de los resultados a fichero
  - Importación del fichero de resultados a BD
- Tiempo total: < 3’. ~1.666 tramas/s

CLASIFICADOR PREDICTIVO
DIAGRAMA DEL MODELADO Y USO



CONCLUSIONES

- ✓ No existe una solución única para un mismo proyecto.
- ✓ Es más importante invertir tiempo en investigar como hacerlo que hacerlo directamente.
- ✓ Las BD NoSQL son ideales para gestionar enormes cantidades de datos.
- ✓ Los resultados del entrenamiento del modelo dependen en gran medida de los datos usados.
- ✓ La cantidad de los datos es importante, pero la calidad lo es más.

FIN

¡Gracias por su atención!