

Sistema Neurosimbólico para la Integridad de Tanques de Almacenamiento

Gaston Pepe

2025-04-17

Sistema Neurosimbólico para la Integridad de Tanques de Almacenamiento en la Industria del Petróleo

En la industria del petróleo, garantizar la **integridad mecánica** de los tanques de almacenamiento es crucial para prevenir fugas, cumplir normativas como **API 653**, y optimizar costos operativos. Presento un sistema **neurosimbólico** que combina inteligencia artificial y razonamiento lógico para evaluar riesgos de corrosión en tanques y recomendar acciones específicas basadas en estándares de la industria.

¿Qué es un sistema neurosimbólico?

Un sistema neurosimbólico integra:

- **Redes neuronales:** Para aprender patrones de datos, como predecir riesgos de corrosión basados en mediciones.
- **Razonamiento simbólico:** Para aplicar reglas lógicas, como decidir inspecciones o reparaciones según la norma API 653.

Este enfoque combina lo mejor de ambos mundos: la capacidad de aprendizaje de la IA con la interpretabilidad del razonamiento basado en reglas.

Caso práctico: Evaluación de riesgos en tanques

Desarrollé un prototipo que:

1. **Predice riesgos de corrosión** usando un modelo de aprendizaje automático en Python.
2. **Toma decisiones** con reglas en LISA (Common Lisp), alineadas con API 653.
3. **Integra ambos componentes** usando py4cl2 para comunicación en tiempo real.

Ejemplo de Salida del software

Si ingresamos los siguientes datos

Espesor de pared (mm)	Temperatura interna (°C)	Años en servicio
Tipo de fluido (ligero o pesado)	Presión interna (psi)	Recubrimiento protector (sí/no)
Protección catódica (sí/no)		
8.0	60	10
1	20	0
0		

Resultado:

Riesgo medio. Programar inspección visual y ultrasónica en 6 meses (API 653, Anexo F - RBI)."

Implementacion prototipo

Componente neuronal (Python)

El modelo utiliza un clasificador Random Forest para predecir el riesgo de corrosión (bajo, medio, alto) basado en:

- Espesor de pared (mm)
- Temperatura interna (°C)
- Años en servicio
- Tipo de fluido (ligero o pesado)
- Presión interna (psi)
- Recubrimiento protector (sí/no)
- Protección catódica (sí/no)

```
from sklearn.ensemble import RandomForestClassifier
import numpy as np
```

```
# Dataset: [thickness (mm), temperature (°C), years, fluid_type (0=light, 1=heavy), p
X = np.array([
    [10.0, 50, 5, 0, 15, 1, 1], [8.0, 60, 10, 1, 20, 0, 0], [12.0, 40, 3, 0, 10, 1, 1]
    [7.0, 70, 15, 1, 25, 0, 0], [9.0, 55, 8, 0, 18, 1, 0], [6.0, 65, 12, 1, 22, 0, 0]
    [11.0, 45, 4, 0, 12, 1, 1], [5.0, 75, 20, 1, 30, 0, 0], [9.5, 50, 7, 0, 15, 1, 1]
    [7.5, 62, 13, 1, 20, 0, 1], [10.5, 48, 6, 0, 14, 1, 0], [6.5, 68, 18, 1, 28, 0, 0]
])
y = np.array([0, 1, 0, 2, 1, 2, 0, 2, 0, 1, 0, 2]) # 0=low, 1=medium, 2=high
```

```

# Train model
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X, y)

def predict_risk(thickness, temperature, years, fluid_type, pressure, coating, cathod
    X_new = np.array([[thickness, temperature, years, fluid_type, pressure, coating,
    risk_level = model.predict(X_new)[0]
    confidence = model.predict_proba(X_new).max()
    risk_map = {0: "low", 1: "medium", 2: "high"}
    return {
"risk": risk_map[risk_level],
"confidence": float(confidence),
"thickness": float(thickness)
    }

```

Componente simbólico (Common Lisp con LISA)

Las reglas en LISA evalúan el riesgo predicho y el espesor de pared, recomendando acciones específicas según **API 653**:

```

(eval-when (:compile-toplevel :load-toplevel :execute)
  (when (not (find-package "TANK-INTEGRITY"))
    (defpackage "TANK-INTEGRITY"
      (:use "COMMON-LISP" "LISA")
      (:export "EVALUATE-TANK"))))

(in-package "TANK-INTEGRITY")

(ql:quickload :py4cl2)
(py4cl2:python-start)
(py4cl2:import-module "tank_integrity")

(make-inference-engine)

(deftemplate tank-risk ()
  (slot risk)
  (slot confidence)
  (slot thickness))

(defun calculate_minimum_thickness (height diameter specific_gravity)
  (/ (* 2.6 height diameter specific_gravity) (* 24800 1.0)))

```

```

(defrule low-risk ()
  (tank-risk (risk ?r) (confidence ?c) (thickness ?t))
  (test (string= ?r "low"))
  (test (> ?c 0.7))
  (test (> ?t (calculate_minimum_thickness 40 100 0.85)))
  =>
  (format t "Low risk. Continue normal operation. External inspection every 5 years (

(defrule medium-risk-inspection ()
  (tank-risk (risk ?r) (confidence ?c) (thickness ?t))
  (test (string= ?r "medium"))
  (test (> ?c 0.7))
  (test (> ?t (calculate_minimum_thickness 40 100 0.85)))
  =>
  (format t "Medium risk. Schedule visual and ultrasonic inspection within 6 months (

(defrule medium-risk-critical-thickness ()
  (tank-risk (risk ?r) (confidence ?c) (thickness ?t))
  (test (string= ?r "medium"))
  (test (> ?c 0.7))
  (test (<= ?t (calculate_minimum_thickness 40 100 0.85)))
  =>
  (format t "Medium risk, critical thickness. Schedule immediate internal inspection

(defrule high-risk ()
  (tank-risk (risk ?r) (confidence ?c) (thickness ?t))
  (test (string= ?r "high"))
  (test (> ?c 0.7))
  =>
  (format t "High risk. Suspend operation and repair immediately (API 653, Section 9.

(defun evaluate-tank (thickness temperature years fluid_type pressure coating cathodi
  (reset)
  (let ((result (py4cl2:python-call "tank_integrity.predict_risk"
    thickness temperature years fluid_type
    pressure coating cathodic_protection)))
    (assert (tank-risk (risk (gethash "risk" result))
      (confidence (gethash "confidence" result))
      (thickness (gethash "thickness" result)))))
    (run)))

;; Example
(evaluate-tank 8.0 60 10 1 20 0 0)

```

Medium risk. Schedule visual and ultrasonic inspection within 6 months (API 653, Anexo F)
Riesgo medio. Programar inspección visual y ultrasónica en 6 meses (API 653, Anexo F)

Beneficios

- **Seguridad:** Predice riesgos de corrosión y recomienda acciones para prevenir fallos.
- **Cumplimiento normativo:** Alineado con API 653, con referencias específicas a secciones de la norma.
- **Eficiencia:** Automatiza la evaluación de riesgos, reduciendo costos de inspecciones manuales.
- **Escalabilidad:** Adaptable a datos reales y otros equipos estáticos.

Conclusión

Este sistema demuestra el potencial de la IA neurosimbólica para resolver problemas críticos en la industria del petróleo. Estoy explorando cómo extenderlo a otros equipos estáticos o integrarlo con datos reales. ¿Qué opinas? ¿Te interesa colaborar en aplicaciones similares?

Conéctate conmigo en LinkedIn para discutir cómo la IA puede transformar la industria del petróleo!

#IA #Petróleo #API653 #Neurosimbólico #Ingeniería