Gallego Fontenla, V., Gamallo Fernández, P., Vidal, J. C., Lama Penín, M., Antelo González, L.: De la detección a la interpretación: Un algoritmo basado en LLMs para la descripción de cambios en el flujo de control. In: Fabra, J. (ed.) Actas de las XXI Jornadas de Ciencia e Ingeniería de Servicios (JCIS 2026). Sistedes (2026). https://hdl.handle.net/11705/JCIS/2026/37

De la detección a la interpretación: Un algoritmo basado en LLMs para la descripción de cambios en el flujo de control

Víctor Gallego-Fontenla[1,2], Laura Antelo González[2], Pedro Gamallo-Fernandez[1,2], Juan C. Vidal[1,2], and Manuel Lama[1,2]

[1] Departamento de Electrónica e Computación, Universidade de Santiago de Compostela (USC), España
[2] Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS), Universidade de Santiago de Compostela (USC), España
{victorjose.gallego, pedro.gamallo.fernandez, juan.vidal, manuel.lama}@usc.gal, laura.antelo.gonzalez@rai.usc.gal

Resumen Los procesos de negocio son entidades dinámicas sujetas a cambios a lo largo del tiempo. Aunque la minería de procesos ofrece diferentes métodos para detectar estos cambios, existe una importante brecha en su identificación y descripción. En este trabajo, proponemos un algoritmo que integra abstracción de trazas, descubrimiento de modelos y LLMs con el objetivo de identificar y explicar los patrones de cambio del flujo de control directamente en lenguaje natural.

Keywords: minería de procesos · concept drift · LLMs · explicabilidad

1. Introducción

La minería de procesos permite a las organizaciones obtener una visión objetiva de su funcionamiento a partir de los registros de eventos. Sin embargo, los procesos evolucionan constantemente en entornos dinámicos, fenómeno conocido como concept drift. Aunque existen algoritmos eficaces para detectar estos cambios [4], la interpretación detallada de estos en lenguaje natural sigue requiriendo un exhaustivo análisis manual experto [3], evidenciando una brecha semántica importante.

Paralelamente, los Grandes Modelos de Lenguaje (LLMs) han demostrado un enorme potencial para el análisis automatizado de procesos [1]. Para adaptar estos modelos a la naturaleza de los registros, la literatura ha explorado diferentes técnicas de abstracción de trazas [2] y optimización de instrucciones (*prompt engineering*).

A pesar de estos avances, la aplicación de LLMs a la interpretación de los cambios permanece inexplorada. En este trabajo proponemos una aproximación para la caracterización de cambios en el flujo de control. Nuestro enfoque busca identificar patrones de cambio y generar descripciones precisas, facilitando a los analistas la comprensión inmediata de la evolución organizativa.

This work is licensed under a Creative Commons “Attribution-NonCommercial-NoDerivatives 4.0 International” license.

Gallego Fontenla, V., Gamallo Fernández, P., Vidal, J. C., Lama Penín, M., Antelo González, L.: De la detección a la interpretación: Un algoritmo basado en LLMs para la descripción de cambios en el flujo de control. In: Fabra, J. (ed.) Actas de las XXI Jornadas de Ciencia e Ingeniería de Servicios (JCIS 2026). Sistedes (2026). https://hdl.handle.net/11705/JCIS/2026/37

2. Explicación de cambios mediante el empleo de LLMs

En este artículo proponemos una metodología de análisis post-detección para traducir automáticamente los cambios en el flujo de control a descripciones en lenguaje natural. Asumiendo que un algoritmo externo ya ha segmentado el registro en periodos estables (pre y post-cambio), nuestro sistema se enfoca exclusivamente en procesar estas particiones para verbalizar las diferencias estructurales. Para lograr este objetivo, proponemos un *pipeline* de procesamiento con dos componentes de diseño claramente diferenciadas: a) la abstracción de los datos de entrada; y b) el diseño de las instrucciones (prompt engineering) proporcionadas al LLM.

2.1. Abstracción de Datos

En nuestra aproximación evaluamos diferentes niveles de abstracción para representar el comportamiento del proceso en los periodos *pre-* y *post-*cambio:

- **Registros**: Como línea base, se proporciona al LLM dos muestras del registro de eventos, una previa y otra posterior al cambio, sin ningún tipo de preprocesamiento, evaluando así la capacidad del LLM para inferir el flujo de control directamente de la secuencia de eventos.
- **Variantes de trazas**: Para reducir la cantidad de información redundante proporcionada al LLM, se agrupan las ejecuciones idénticas, conservando únicamente la secuencia de actividades y la frecuencia de cada variante. Así, evitamos proporcionar casos individuales al LLM, lo que puede ayudar a evitar que haga comparaciones por pares.
- **DFG**:Se proporciona al LLM un modelo que captura las relaciones de precedencia directa entre actividades y sus frecuencias, ofreciendo una representación mucho más compacta y centrada en las dependencias causales entre actividades.

2.2. Diseño de las instrucciones

Para maximizar la precisión de las descripciones generadas y mitigar las alucinaciones del modelo al identificar los patrones de cambio, hemos diseñado y evaluado tres estrategias de *prompting* con diferente complejidad:

- ***Zero-Shot***: Solicitamos al modelo que describa las diferencias estructurales entre las dos versiones del proceso proporcionadas sin incluir ejemplos ni instrucciones de razonamiento. Esto permite establecer una línea base a partir de la cual proponer mejoras.
- ***Factuality***: Introducimos restricciones estrictas en la consulta para obligar al modelo a fundamentar su respuesta única y exclusivamente en las observaciones que pueda extraer de los datos proporcionados. Así, priorizamos la adaptación al caso de uso frente al conocimiento genérico sobre procesos que pueda tener el LLM.

This work is licensed under a Creative Commons “Attribution-NonCommercial-NoDerivatives 4.0 International” license.

Gallego Fontenla, V., Gamallo Fernández, P., Vidal, J. C., Lama Penín, M., Antelo González, L.: De la detección a la interpretación: Un algoritmo basado en LLMs para la descripción de cambios en el flujo de control. In: Fabra, J. (ed.) Actas de las XXI Jornadas de Ciencia e Ingeniería de Servicios (JCIS 2026). Sistedes (2026). https://hdl.handle.net/11705/JCIS/2026/37


```
ROLE: You are an expert process mining analyst specialized in control-flow and structural change explanation
CONTEXT:
    - You receive two Directly-Follows Graphs (DFGs) from the same process.
    - 'Original DFG' is the baseline. 'Modified DFG' is the updated version.
    - The DFGs represent transitions between activities in the format 'Activity A -> Activity B (freq. N)'.
CHANGE CLASSIFICATION CODES:
    You must classify the structural change using strictly ONE of the following codes:
        - cm: Move fragment into/out of conditional branch
        - cp: Duplicate fragment
        ...
TASK:
    1) Carefully compare the transitions between the Original DFG and the Modified DFG.
    2) Identify what specific activities or paths have changed.
    3) Map the observed change to exactly ONE of the classification codes provided above.
OUTPUT FORMAT: Return a response with two exact sections:
    1. [[ ## reasoning ## ]]: A step-by-step analysis of the DFG edges comparing A and B.
    2. [[ ## json ## ]]: A STRICT JSON object with the following schema:
        {
            "change_code": "<One of the 11 codes>",
            "description": "<Concise explanation of what structurally changed>",
            "evidence": ["<Edge X -> Y present in A but not B>", ...],
        }
```

Figura 1. Ejemplo de consulta *Chain of Thought*

```
[[ ## reasoning ## ]]
    1. **Identify common edges**: Most edges exist in both DFGs. Their frequencies have increased but the
    structure is unchanged.
    2. **Detect added edges**: The Modified DFG contains two edges that are **absent** from the Original DFG
    : ...
    3. **Interpret the added edges**: Both new edges point **backwards** in the process flow: ...
    4. **Map to change classification**: Introducing a back edge that makes a fragment executable more than
    once corresponds to **make fragment loopable**, which is coded as **lp**.
    5. **Select the single appropriate code**: No other structural transformation (parallelism, conditional
    branching, duplication, etc.) is evident; the key structural change is the newly created loop.
[[ ## json ## ]]
    {
        "change_code": "lp",
        "description": "Edges from Asses_elegibility back to Check_credit_history and Appraise_property were
         added, turning the credit check/property appraisal fragment into a loopable segment.",
        "evidence": [
            "Asses_elegibility -> Check_credit_history present in Modified DFG but not in Orig. DFG",
            "Asses_elegibility -> Appraise_property present in Modified DFG but not in Original DFG"
        ],
    }
```

Figura 2. Ejemplo de respuesta

- ***Chain of Thought***: Inducimos al modelo a generar un razonamiento paso a paso antes de emitir una conclusión, pidiéndole que descomponga la tarea en diferentes pasos, lo que mejora su lógica deductiva. Además, suministramos al modelo un catálogo predefinido de patrones de cambio [5] con una descripción breve de cada uno. La Figura 2 muestra un ejemplo de consulta de este tipo.

3. Evaluación

Los resultados preliminares de nuestra aproximación, ilustrados en la **??**, revelan que la calidad de la descripción depende directamente del preprocesamiento. Así, los mejores resultados se obtienen empleando las abstracciones más avanzadas. Sin ellas, el LLM tiende a comparar los registros traza a traza, destacando cambios entre pares en lugar de generalizar. Asimismo, constatamos un claro compromiso: los prompts más complejos incrementan el consumo de *tokens*, pero logran descripciones notablemente superiores al adaptar mejor al modelo a la tarea. A pesar de esto, los patrones de cambio que involucran paralelismos o decisiones presentan problemas en la detección debido a las limitaciones intrínsecas de las abstracciones probadas hasta el momento para representar adecuadamente dichas estructuras.

This work is licensed under a Creative Commons “Attribution-NonCommercial-NoDerivatives 4.0 International” license.

Gallego Fontenla, V., Gamallo Fernández, P., Vidal, J. C., Lama Penín, M., Antelo González, L.: De la detección a la interpretación: Un algoritmo basado en LLMs para la descripción de cambios en el flujo de control. In: Fabra, J. (ed.) Actas de las XXI Jornadas de Ciencia e Ingeniería de Servicios (JCIS 2026). Sistedes (2026). https://hdl.handle.net/11705/JCIS/2026/37

4. Conclusiones y trabajo futuro

En este trabajo hemos presentado una aproximación a la interpretación del *concept drift* en procesos mediante LLMs. Los experimentos revelan dos hallazgos: primero, proveer más información en el *prompt* reduce alucinaciones y mejora la identificación de patrones y actividades; y segundo, abstraer los registros de eventos resulta crucial para evitar comparaciones "traza a traza", focalizando al modelo en su tarea.

Como trabajo futuro, planeamos optimizar la representación de datos para reducir el consumo de *tokens*, proporcionando al LLM únicamente la diferencia directa entre modelos pre y post-cambio. Además, evaluaremos el uso de modelos declarativos, que pueden integrarse mejor con el lenguaje natural. Finalmente, planeamos ampliar el marco experimental combinando las métricas cuantitativas actuales con validaciones cualitativas por parte de expertos.

Agradecimientos

Este trabajo ha sido parcialmente financiado por la Agencia Estatal de Investigación (proyectos PID2023-149549NB-I00 y PDC2025-166312-I00), la Consellería de Educación, Ciencia, Universidades e Formación Profesional de la Xunta de Galicia (acreditación como Centro de Investigación de Galicia 2024-2027 ref. ED431G-2023/04) y la Unión Europea (Fondo Europeo de Desarrollo Regional-FEDER).

Referencias

1. Berti, A., Kourani, H., Häfke, H., Li, C., Schuster, D.: Evaluating large language models in process mining: Capabilities, benchmarks, and evaluation strategies. In: Proceedings of the 25th International Conference on Business Process Modeling, Development and Support (BPMDS 2024). pp. 13–21. Springer (2024). https://doi.org/10.1007/978-3-031-61007-3_2
2. Berti, A., Schuster, D., van der Aalst, W.M.P.: Abstractions, Scenarios, and Prompt Definitions for Process Mining with LLMs: A Case Study. In: Revised Selected Papers of the 2023 International Business Process Management Workshops (BPM 2023). pp. 427–439. Springer (2023). https://doi.org/10.1007/978-3-031-50974-2_32
3. Fontenla-Seco, Y., Lama, M., González-Salvado, V., Peña-Gil, C., Diz, A.J.B.: A Framework for the Automatic Description of Healthcare Processes in Natural Language: Application in an Aortic Stenosis Integrated Care Process. Journal of Biomedical Informatics **128**, 104033 (2022). https://doi.org/10.1016/J.JBI.2022.104033
4. Sato, D.M.V., De Freitas, S.C., Barddal, J.P., Scalabrin, E.E.: A Survey on Concept Drift in Process Mining. ACM Computing Surveys **54**, 189:1–189:38 (2021). https://doi.org/10.1145/3472752
5. Weber, B., Reichert, M., Rinderle-Ma, S.: Change Patterns and Change Support Features - Enhancing Flexibility in Process-aware Information Systems. Data and Knowledge Engineering **66**, 438–466 (2008). https://doi.org/10.1016/j.datak.2008.05.001

This work is licensed under a Creative Commons "Attribution-NonCommercial-NoDerivatives 4.0 International" license.