

# Optimización de la Generación de Chatbots Mediante Técnicas de Líneas de Producto

José A. Zamudio<sup>1</sup>[0000-0002-5025-7424], Jordi Cabot<sup>2</sup>[0000-0003-2418-2489], and  
David Benavides<sup>1</sup>[0000-0002-8449-3273]

<sup>1</sup> Universidad de Sevilla, Dept. Lenguajes y Sistemas Informáticos, Av. Reina  
Mercedes s/n Sevilla - España {[jzamudio](mailto:jzamudio@us.es), [benavides](mailto:benavides@us.es)}@us.es

<sup>2</sup> Luxembourg Institute of Science and Technology, Maison de l'innovation 5, Avenue  
des Hauts-Fourneaux L-4362 Esch-sur-Alzette [jordi.cabot@list.lu](mailto:jordi.cabot@list.lu)

**Resumen** Los chatbots han ganado cada vez más popularidad en diversos sectores, ya que proporcionan un medio para automatizar las interacciones con los participantes, reduciendo la carga de trabajo y mejorando la eficiencia. Sin embargo, desarrollar un chatbot con un alto nivel de precisión y personalización puede llevar mucho tiempo y, en la actualidad, no existe un proceso sistemático para reutilizar de forma eficiente la información de dominio para agilizar este proceso. Para superar este reto, proponemos una solución que aplica técnicas de ingeniería de línea de producto sobre el proceso de generación de chatbot para generar familias de chatbots que comparten características entre sí, incluidas las intenciones del usuario, los agentes de procesamiento del lenguaje natural y otras características relevantes, para integrarlos en los chatbots. Además, se muestra una herramienta que aplica los conceptos discutidos en este proyecto y que ofrece una solución práctica que podría servir para mejorar la eficacia y la eficiencia del desarrollo de chatbots.

**Keywords:** software product lines · chatbots · artefact reusability

## 1. Introducción

Los chatbots simulan conversaciones con los usuarios que utilizan técnicas de Procesamiento del Lenguaje Natural (PLN) [5] para comprender las entradas del usuario y generar respuestas adecuadas. Herramientas como Dialogflow o Rasa permiten definir y desplegar chatbots en plataformas de mensajería instantánea. No obstante, resulta necesario contextualizar la creación de los chatbots. Esta tarea se vuelve aún más laboriosa en el caso de chatbots que comparten información. Introducir manualmente la misma información exige modificar y volver a entrenar a cada uno de ellos de forma individual. Si, por ejemplo, nos encargáramos de crear chatbots para una cadena de restaurantes, tendríamos que abordar la tarea manualmente, creando tanto la estructura como proporcionando información al chatbot uno a uno. Para ello, las Líneas de Producto Software (SPL) ofrecen técnicas y métodos para crear productos que comparten características comunes [1]. Estos métodos se utilizan a menudo para definir familias, lo que



permite la representación de toda una gama de productos utilizando un único modelo de información conocido como modelo de características [4]. De esta manera, la definición de una línea de producto chatbot nos permite procesar la información una sola vez para generar chatbots seleccionando únicamente las características que deseemos incorporar. Este artículo presenta Xatkit-SPL<sup>3</sup>, una herramienta para generar chatbots empleando técnicas de líneas de producto y utilizando información específica de dominio. El artículo explica el funcionamiento de la herramienta, destaca su contribución e incluye los retos futuros que deben abordarse para mejorar la funcionalidad de la herramienta.

## 2. Xatkit-SPL

Xatkit-SPL es una herramienta que atiende a dos roles de usuario: Expertos y Usuarios. Los Expertos aportan conocimiento del dominio para crear una familia de chatbots, mientras que los Usuarios interactúan con el chatbot generado. En la Figura 1 podemos ver cómo están conectados los diferentes módulos y roles.

### 2.1. Arquitectura

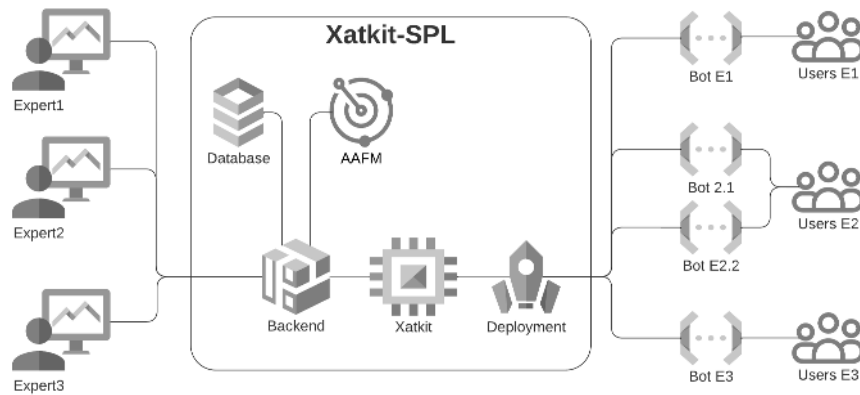


Figura 1. Componentes de Xatkit-SPL

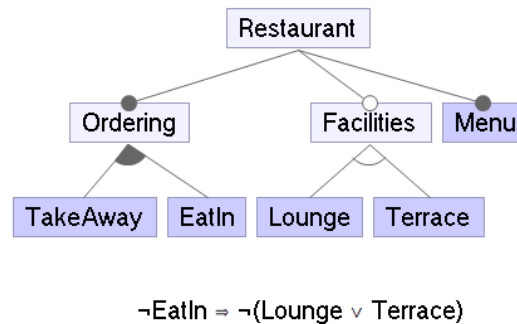
1. Backend: permite a los expertos definir las líneas de producto y la información de dominio necesaria para generar chatbots.
2. Database: facilita la gestión de usuarios y cambios para los chatbots y las líneas de productos.

<sup>3</sup> Repositorio de Xatkit-SPL: <https://github.com/joszamama/xatkit-spl-api>

3. AAFM [3]: proporciona un mecanismo de validación para asegurar que las características del chatbot son consistentes y compatibles con el modelo de características definido. Para ello utilizamos FLAMA<sup>4</sup>.
4. Builder: genera chatbots basados en las características seleccionadas y la información del dominio utilizando el DSL [2] de Xatkit<sup>5</sup>.
5. Deployment: despliega los chatbots para su acceso público.

## 2.2. Funcionamiento

Para generar nuestra familia de chatbots debemos aportar un modelo de características UVL [6]. En la Figura 2 podemos ver un ejemplo de modelo para el dominio de la restauración, con características como menú, distintas formas de pedir e instalaciones. Para cada característica identificada en el modelo, el experto en el dominio debe proporcionar un conjunto de preguntas que el usuario puede hacer sobre esa característica y su correspondiente respuesta. Cada característica se convierte en un conjunto de intenciones, que son preguntas sobre la característica correspondiente y sus respuestas asociadas. De esta manera, el experto puede seleccionar qué características incluir en los chatbots que desea generar seleccionando las intenciones generadas por Xatkit-SPL. Esta petición se analiza con FLAMA, que valida la selección de intenciones del usuario para garantizar que no infringe ninguna restricción del modelo. El proceso concluye con la descarga de un archivo Dockerfile, que puede ser lanzado y desplegado para permitir el acceso al chatbot. Todos los chatbot generados están equipados con un servidor NLP propio, de código abierto y personalizable que permite la comprensión conversacional. El proceso completo descrito en esta sección se muestra en el vídeo demostración<sup>6</sup>.



**Figura 2.** FM de una línea de producto de chatbots para restaurantes

<sup>4</sup> Repositorio de FLAMA: <https://flamapy.github.io/>

<sup>5</sup> Repositorio de Xatkit: <https://github.com/xatkit-bot-platform>

<sup>6</sup> Video Demo: <https://youtu.be/2mnehWpxWI>

### 3. Conclusiones y trabajo futuro

Mediante el uso de técnicas de líneas de producto, hemos desarrollado una nueva herramienta que permite la generación de chatbots, reutilizando información de dominio y que incorpora herramientas de investigación y desarrollo, como FLAMA y Xatkit, y cuenta con una API que facilita su integración. La investigación presenta múltiples desafíos. En primer lugar, resulta complejo definir las características para un mismo modelo, ya que existen diversas formas de hacerlo. Mientras que una estrategia podría consistir en definir las intenciones como características concretas del modelo, un enfoque más sofisticado podría definir una máquina de estados basada en intenciones. En segundo lugar, mantener la información actualizada es un reto crucial. Por ejemplo, si la hora de cierre de un restaurante cambia, actualizar la línea de producto es un proceso sencillo. No obstante, actualizar con la nueva información todos los chatbots previamente generados sería una tarea tediosa y poco práctica. En consecuencia, es necesario un mecanismo automatizado que permita actualizar los chatbots existentes cuando se produzcan cambios en la línea de producto.

**Agradecimientos** This work was partially supported by FEDER/Ministry of Science and Innovation/Junta de Andalucía/State Research Agency with the following grants: *Data-pl*(PID2022-138486OB-I00), TASOVA PLUS research network (RED2022-134337-T) and *METAMORFOSIS* (FEDER\_US-1381375)

### Referencias

1. Benavides, D., Segura, S., Ruiz-Cortés, A.: Automated analysis of feature models 20 years later: A literature review. *Information Systems* **35**(6), 615–636 (2010)
2. Daniel, G., Cabot, J., Deruelle, L., Derras, M.: Xatkit: a multimodal low-code chatbot development framework. *IEEE Access* **8**, 15332–15346 (2020)
3. Galindo, J.A., Benavides, D.: A python framework for the automated analysis of feature models: A first step to integrate community efforts. In: *Proceedings of the 24th acm international systems and software product line conference-volume b*. pp. 52–55 (2020)
4. Lee, K., Kang, K.C., Lee, J.: Concepts and guidelines of feature modeling for product line software engineering. In: *Software Reuse: Methods, Techniques, and Tools: 7th International Conference, ICSR-7 Austin, TX, USA, April 15–19, 2002 Proceedings* 7. pp. 62–77. Springer (2002)
5. Nadkarni, P.M., Ohno-Machado, L., Chapman, W.W.: Natural language processing: an introduction. *Journal of the American Medical Informatics Association* **18**(5), 544–551 (2011)
6. Sundermann, C., Feichtinger, K., Engelhardt, D., Rabiser, R., Thüm, T.: Yet another textual variability language? a community effort towards a unified language. In: *Proceedings of the 25th ACM International Systems and Software Product Line Conference-Volume A*. pp. 136–147 (2021)

