

Wedding Organizer: Sistema de Planificación de Bodas

Lia S. López Rosales and Ariadna Velázquez Rey

Facultad de Matemática y Computación (MATCOM)
UH, La Habana

https://github.com/AriadnaVelazquez744/Events_Organizer_Helper.git

Abstract. Se presenta *Wedding Organizer* (Organizador de Bodas) un sistema multi-agente inteligente para la planificación automatizada de eventos sociales, especializado en bodas. El sistema implementa una arquitectura BDI (*Beliefs* (Creencias) - *Desires* (Deseos) - *Intentions* (Intenciones))[1] integrada con sistemas locales especializados de *Retrieval-Augmented-Generation* (RAG (Generación Aumentada por Recuperación)) [2] para proporcionar un plan personalizado de recomendaciones de local de celebración (*venue*), servicio proveedor de gastronomía (*catering*) y decoración. Se evaluó la efectividad del sistema mediante experimentos estadísticos automatizados, obteniendo una tasa de significancia del 63.2% en 19 análisis realizados, con una potencia promedio de 0.640 y un tamaño de efecto medio de 0.285. Los resultados demuestran la viabilidad del enfoque propuesto para automatizar procesos de planificación de eventos sociales.

Keywords: Sistemas Multi-Agente, Arquitectura BDI, RAG, Planificación de Eventos, Inteligencia Artificial, ANOVA

1 Introducción

La planificación de eventos sociales, particularmente bodas, representa un desafío complejo que involucra múltiples decisiones interdependientes, restricciones presupuestarias y preferencias personalizadas. Tradicionalmente, este proceso requiere la intervención de planificadores profesionales que coordinan múltiples proveedores y gestionan numerosos detalles simultáneamente. Sin embargo, este enfoque manual puede presentar limitaciones en términos de escalabilidad, consistencia y personalización.

El proyecto Wedding Organizer aborda esta problemática mediante la implementación de un sistema multi-agente inteligente que automatiza y optimiza el proceso de planificación de eventos en aspectos imprescindibles del mismo (*venue*, *catering*, decoración). El sistema se fundamenta en la arquitectura BDI (*Beliefs-Desires-Intentions*) como agente central planificador, integrada con sistemas locales especializados de *Retrieval-Augmented Generation* (RAG) en agentes de decisión para proporcionar recomendaciones contextualmente relevantes y personalizadas.

La motivación principal del proyecto radica en la necesidad de desarrollar soluciones tecnológicas que puedan asistir a los usuarios (clientes o profesionales) en la toma de decisiones complejas, reduciendo la carga cognitiva y mejorando la calidad de las decisiones finales. El enfoque multi-agente permite la especialización de responsabilidades y la coordinación efectiva entre diferentes aspectos de la planificación.

2 Descripción del Problema

La planificación de bodas presenta características que la convierten en un problema complejo de optimización multi-objetivo:

- **Multi-dimensionalidad:** Involucra múltiples categorías (venue, catering, decoración) con criterios específicos y restricciones propias.
- **Interdependencia:** Las decisiones en una categoría pueden afectar las opciones disponibles en otras.
- **Restricciones presupuestarias:** Requiere distribución óptima de recursos financieros limitados.
- **Preferencias subjetivas:** Incorpora criterios cualitativos difíciles de cuantificar.
- **Incertidumbre:** Los datos de proveedores pueden ser incompletos o inconsistentes.

Estos factores generan un espacio de búsqueda exponencial que resulta complicado en enfoques tradicionales de optimización. Además, la naturaleza dinámica del mercado de servicios para eventos requiere sistemas capaces de adaptarse a cambios en disponibilidad, precios y ofertas.

3 Solución Implementada

3.1 Arquitectura General del Sistema

El sistema se estructura en torno a una arquitectura multi-agente distribuida que implementa un agente principal BDI dedicado al control del flujo de decisiones y 4 agentes especializados en tareas de decisión en diferentes dominios. Los agentes implementados son:

- **PlannerAgent:** Coordinador principal de decisiones que implementa el ciclo BDI completo.
- **VenueAgent:** Especializado en búsqueda y recomendación de venues.
- **CateringAgent:** Gestiona opciones de catering y servicios gastronómicos.
- **DecorAgent:** Maneja recomendaciones de decoración y diseño (principalmente arreglos florales).
- **BudgetDistributorAgent:** Optimiza la distribución de un presupuesto total.

Se utilizó esta estructura por su capacidad de coordinar múltiples decisiones en dominios distintos de forma eficiente.

3.2 Implementación del Ciclo BDI

El ciclo BDI se implementa en el PlannerAgent, siguiendo las especificaciones teóricas de Rao y Georgeff:

Beliefs (Creencias) Las creencias del sistema se estructuran en un esquema jerárquico que incluye:

- Criterios del usuario (presupuesto, número de invitados, estilo).
- Estado del sistema (tareas pendientes, completadas, errores).
- Distribución de presupuesto asignada.
- Resultados de búsquedas previas.
- Historial de errores y estrategias de corrección.

Desires (Deseos) Los deseos se generan dinámicamente basados en la petición del usuario:

- Desire principal: Completar la planificación del evento.
- Desires específicos: Encontrar venue, catering, decoración óptimos y distribuir presupuesto.
- Desires de corrección: Resolver conflictos y errores detectados.

Intentions (Intenciones) Las intenciones se crean a partir de los deseos activos y se traducen en planes de acción concretos:

- Distribución de presupuesto.
- Búsquedas especializadas por categoría.
- Estrategias de corrección y recuperación de errores.

La arquitectura BDI permite tener agentes controladores que reacciona y toma decisiones inteligentes en dependencia del estado del sistema y las acciones de otros agentes.

3.3 Sistema RAG Integrado

Cada agente especializado implementa un sistema RAG que combina:

- **Base de Conocimiento:** Grafos de conocimiento específicos por dominio.
- **Motor de Recuperación:** Búsqueda semántica y por similitud.
- **Componente de Generación:** Generación de recomendaciones contextuales.

El sistema RAG se optimiza para manejar datos heterogéneos e incompletos, implementando estrategias de enriquecimiento dinámico que mejoran la calidad de las recomendaciones.

3.4 Crawler Inteligente con Enriquecimiento Dinámico

Se implementó un sistema de crawling avanzado que incluye:

- **Validación de Calidad:** Evaluación automática de la completitud y consistencia de datos.
- **Enriquecimiento Dinámico:** Búsqueda automática de información faltante en fuentes externas.
- **Monitoreo Continuo:** Seguimiento de frescura y calidad de la información existente.

Resulta necesario que el sistema sea capaz de actualizar su conocimiento con grandes cantidades de datos (de fuentes confiables) regularmente, si se detectan pocos datos actuales se envían alertas y es posible activar fase estática y encontrar información más actual.

4 Consideraciones de Implementación

4.1 Arquitectura de Comunicación

Se adoptó un patrón de *MessageBus* (Bus de Mensajes) para facilitar la comunicación asíncrona entre agentes, permitiendo:

- Desacoplamiento entre componentes.
- Escalabilidad horizontal.
- Tolerancia a fallos.
- Trazabilidad completa de mensajes.

Este patrón permite que el sistema sea capaz de cambiar de dominio específico (bodas) o ampliar su especialidad en él con solo cambios mínimos en el Agente de Planificación.

4.2 Gestión de Memoria y Estado

Se implementó un sistema de gestión de memoria de sesión que mantiene:

- Estado persistente de creencias por sesión.
- Historial de decisiones y justificaciones.
- Mecanismos de recuperación ante fallos.

Permite la implementación de un sistema de retroalimentación (feedback) y cambios sobre una propuesta generada para el usuario.

4.3 Manejo de Errores y Recuperación

El sistema incluye estrategias de manejo de errores:

- Detección automática de errores en tareas.
- Estrategias de corrección basadas en RAG.
- Reintentos inteligentes con backoff exponencial.
- *Fallback* (Mecanismo de Contingencia) a alternativas cuando las opciones principales fallan.

5 Justificación Teórico-Práctica

5.1 Fundamentación de la Arquitectura BDI

La elección de la arquitectura BDI se justifica por su capacidad para modelar comportamientos racionales en agentes autónomos. Según Wooldridge [3], esta arquitectura proporciona:

- **Autonomía:** Los agentes operan sin intervención externa continua.
- **Reactividad:** Respuesta apropiada a cambios en el entorno.
- **Pro-actividad:** Iniciativa en la persecución de objetivos.
- **Socialidad:** Interacción efectiva con otros agentes.

Características esenciales a la hora de automatizar la planificación de eventos.

5.2 Integración RAG especializado

La integración de sistemas RAG especializados se fundamenta en la necesidad de enriquecer conocimiento estructurado con características no explícitas que pueden mejorar la experiencia. Lewis et al. demuestran que esta combinación mejora significativamente la calidad y relevancia de las respuestas en dominios específicos.

5.3 Enfoque Multi-Agente

La adopción de un enfoque multi-agente se justifica por la naturaleza distribuida del problema de planificación de eventos. Según Ferber [4], los sistemas multi-agente ofrecen:

- **Especialización:** Cada agente se enfoca en un dominio específico.
- **Paralelismo:** Procesamiento concurrente de diferentes aspectos.
- **Robustez:** Fallos en un agente no comprometen el sistema completo.
- **Escalabilidad:** Fácil adición de nuevos agentes especializados.

6 Evaluación Cuantitativa y Cualitativa

6.1 Metodología Experimental

Se diseñó un conjunto completo de experimentos estadísticos que incluye:

- **Experimento 1:** Efectividad del sistema BDI.
- **Experimento 2:** Precisión de los sistemas RAG.
- **Experimento 3:** Rendimiento del sistema (latencia, escalabilidad, uso de recursos).
- **Experimento 4:** Efectividad de integración de componentes del sistema.

Cada experimento utiliza metodología estadística con:

- Generación de datos sintéticos controlados.
- Análisis de varianza (ANOVA) y pruebas no paramétricas.
- Cálculo de potencia estadística y tamaño de efecto.
- Validación cruzada de resultados.

6.2 Resultados Cuantitativos

Los experimentos arrojaron los siguientes resultados:

- **Tasa de Significancia:** 63.2% (12 de 19 análisis con conclusiones significativas)
- **Potencia Promedio:** 0.640 (buena potencia estadística)
- **Tamaño de Efecto Promedio:** 0.285 (efecto medio)
- **Tiempo de Respuesta Promedio por Experimento:** 2.3 segundos por recomendación
- **Precisión RAG:** 78.5% en recomendaciones relevantes

6.3 Análisis por Experimento

Efectividad del sistema BDI

- **Resultado:** 2 de 4 análisis significativos (50%).
- **Fortaleza:** Correlaciones BDI muy fuertes ($p \leq 0.001$).
- **Área de mejora:** Reconsideración de intentions.

Precisión de los sistemas RAG

- **Resultado:** 3 de 5 análisis significativos (60%).
- **Fortaleza:** Alta precisión en recomendaciones por tipo.
- **Área de mejora:** Adaptabilidad a patrones complejos.

Rendimiento del sistema

- **Resultado:** 5 de 5 análisis significativos (100%).
- **Fortaleza:** Buena escalabilidad y eficiencia.
- **Logro:** Todas las métricas de rendimiento superaron expectativas.

Efectividad de integración de componentes del sistema

- **Resultado:** 2 de 5 análisis significativos (40%).
- **Fortaleza:** Buena coordinación entre agentes.
- **Área de mejora:** Optimización de flujos de comunicación.

Se pueden consultar más detalles de los resultados en la sección de resultados de experimentos en el proyecto.

7 Resultados y Posibles Mejoras

7.1 Logros Alcanzados

El proyecto logró implementar exitosamente:

- **Arquitectura BDI Completa:** Implementación del ciclo BDI con manejo de creencias, deseos e intenciones.
- **Sistemas RAG Especializados:** Integración efectiva de RAG en múltiples dominios.
- **Crawler Inteligente:** Sistema de enriquecimiento dinámico de datos.
- **Evaluación Estadística Rigurosa:** Mecanismo de evaluación y validación del sistema.
- **Integración Multi-Agente:** Coordinación efectiva entre agentes especializados

7.2 Insuficiencias Identificadas

Limitaciones Técnicas

- **Escalabilidad de Datos:** El sistema actual maneja datasets limitados; se requiere optimización para grandes volúmenes.
- **Latencia de Comunicación:** Los tiempos de respuesta entre agentes pueden optimizarse.
- **Precisión RAG:** Aunque aceptable, la precisión del 78.5% puede mejorarse.

Limitaciones Funcionales

- **Cobertura de Dominio:** El sistema se enfoca en bodas; se requiere extensión a otros tipos de eventos y categorías.
- **Personalización:** Los algoritmos de personalización deben refinarse.

7.3 Propuestas de Mejora

Mejoras Técnicas Inmediatas

- **Optimización y Aumento de Base de Datos:** Aumentar volumen de información y optimizar búsqueda en las bases de conocimiento.
- **Paralelización:** Aprovechar procesamiento paralelo para búsquedas concurrentes.
- **Algoritmos de Machine Learning:** Refinar algoritmos de personalización.
- **ampliar Dominio:** Agregar más características al sistema de decisión (Fecha, Música, etc).

Mejoras Funcionales a Mediano Plazo

- **Expansión de Dominios:** Extender el sistema a otros tipos de eventos (corporativos, sociales).

8 Conclusiones

El proyecto Wedding Organizer demuestra la viabilidad de aplicar sistemas multi-agente BDI integrados con RAG en la automatización de procesos complejos de planificación de eventos, para aumentar productividad en esta área. Los resultados experimentales, con una tasa de significancia del 63.2% y potencia estadística adecuada, validan la efectividad del enfoque propuesto. Aunque se identificaron áreas de mejora, particularmente en escalabilidad y precisión, el sistema proporciona una base sólida para futuro desarrollo en el dominio de la planificación inteligente de eventos. Las propuestas de mejora establecen una hoja de ruta clara para la evolución del sistema hacia una solución de producción robusta y escalable.

9 Trabajo Futuro

Las líneas de investigación futuras incluyen:

- **Optimización Multi-Objetivo:** Implementar algoritmos de optimización multi-objetivo para balancear múltiples criterios de decisión.
- **Aprendizaje por Refuerzo:** Integrar técnicas de aprendizaje por refuerzo para mejorar la toma de decisiones.
- **Análisis de Sentimientos:** Incorporar análisis de sentimientos para evaluar satisfacción del usuario.

References

1. Rao, A.S., Georgeff, M.P.: BDI agents: from theory to practice. In: Proceedings of the First International Conference on Multiagent Systems, pp. 312–319. AAAI Press (1995)
2. Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., Zettlemoyer, L.: Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems* 33, 9459–9474 (2020)
3. Wooldridge, M.: *An Introduction to MultiAgent Systems*. John Wiley & Sons (2009)
4. Ferber, J.: *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*. Addison-Wesley (1999)
5. Georgeff, M.P., Pell, B., Pollack, M.E., Tambe, M., Wooldridge, M.: The belief-desire-intention model of agency. In: *International Workshop on Agent Theories, Architectures, and Languages*, pp. 1–10. Springer (1998)
6. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. *Advances in Neural Information Processing Systems* 30 (2017)