

Validación y Verificación de sistemas basados en el conocimiento

Principales errores en el desarrollo de un sistema basado en el conocimiento.

Los principales errores los podemos encontrar en:

Conocimiento: Errores en el conocimiento experto, tales como conocimiento incorrecto e incompleto.

Ingeniero del Conocimiento: Errores semánticos de significados entre el ingeniero de conocimiento y el especialista. Obtención incompleta del conocimiento proveniente del experto.

Base del Conocimiento: Errores de sintaxis. Errores de contenido, debido a un conocimiento incorrecto e incompleto y a la incertidumbre en las reglas y los hechos.

Motor de inferencia: Errores en la programación. Errores de lógica.

Calidad de un sistema experto.

Un sistema experto de calidad cuenta con las siguientes características:

- **Conclusiones correctas.**
- **Desempeño adecuado.**
- **Código comprensible y comentado.**
- Conclusiones completas.
- Conclusiones congruentes.
- Confiable respecto a la conclusión.
 - Disponibilidad.
- Base de conocimiento verificada.

Funcionalidad de un SBC

La funcionalidad del SBC es la capacidad del sistema para hacer el trabajo para el cual fue destinado.

- Debe cubrir las expectativas para lo que fue construido.
- Debe ser confiable respecto a su funcionamiento.
- Debe explicar sus respuestas.
- Debe permitir que se añada o modifique su conocimiento fácilmente.

Eficiencia y error de un SBC

- Un sistema basado en el conocimiento se considera correctamente validado cuando cubre los requisitos de calidad y funcionalidad, con lo cual aseguramos la aceptabilidad del SBC.
- Un SBC aceptable es un sistema “completo” y eficiente.
- Si en el proceso de verificación y validación se comete errores, esto nos trae como consecuencia un SBC mal estructurado, y por lo tanto un sistema con errores, “incompleto”.

Eficiencia y error de un SBC

- Para que un sistema experto sea herramienta efectiva, los usuarios deben interactuar de una forma fácil, reuniendo dos capacidades para poder cumplirlo:
 - Explicar sus razonamientos y la base de conocimiento:
 - Adquisición de nuevos conocimientos o integrador del sistema: son mecanismos de razonamiento que sirven para modificar los conocimientos anteriores.

Diferencias con Ing. Software

- IS: Ejecución de casos de prueba.
- IC:
 - Criterios para medir su éxito no son objetivos.
 - Se tolera incertidumbre y subjetividad.
 - No se pueden probar fácilmente: grandes espacios de búsqueda.
 - En muchos casos no existen respuestas 'absolutamente correctas' para evaluar el sistema

Verificación

- Construir el sistema correctamente.
- Descubrir y corregir errores en el SBC de naturaleza técnica (el sistema y/o el conocimiento no resulta coherente, en esta fase no se analiza si es válido, sino coherente)
- ¿Quién la realiza? Ingeniero del conocimiento. El experto participará en la fase de validación donde se analiza si el conocimiento es válido y suficiente para el problema abordado.)
- Criterios a verificar en una SBC:
 - Consistencia: no se llega a conclusiones incoherentes
 - Corrección: hay corrección en la sintaxis, no hay errores morfológicos.
 - Completitud: no hay lagunas en capacidad deductiva (casos donde el sistema no respondería y que se puedan dar en la práctica)

Tipos de inconsistencia

- Estructural: Si no contiene ninguna regla inútil: inalcanzable, un callejón sin salida, no ejecutable o redundante:
 - Duplicación: $p \text{ y } q \rightarrow r$; $q \text{ y } p \rightarrow r$,
 - No disparables: $p \text{ y } \neg p \rightarrow r$;
 - Ciclos de reglas: $p \rightarrow r$; $r \rightarrow x$; $p \rightarrow x$ (muy típico con reglas que modifica su parte antecedente)
- Lógica:
 - Reglas con conclusiones o antecedentes redundantes.
 - Subsunción de reglas (reglas que ya están incluidas en otras y si se quitan el sistema funciona igual).
 - Reglas con conclusiones que producen contradicción lógica, y que se podrían ejecutar en una misma situación. Ojo, puede haber inconsistencia teórica pero que no se da en la práctica

Tipos de inconsistencias

- Semántica: Valores ilegales en variables. Se precisa un modelo de coherencia para definir conflictos semánticos porque dependen del contexto
 - Coherencia de un patrón: (Mesa velocidad 38)
 - Coherencia de un conjunto de patrones: (Mesa1 ancho 38)(Mesa1 ancho 44)
 - Coherencia de una regla: Si (Mesa1 ancho 38) Entonces (Mesa1 ancho 44)
 - Coherencia de un conjunto de reglas: Encadenamiento de reglas que lleva a afirmar un conjunto de patrones incoherentes

Definición de Validación

- Construir el sistema correcto.
- Actividad 'viva' no sobre el papel.
- Según ANSI/IEEE 'evaluar la conformidad con la especificación de requisitos'
- En IC: Determinar si el sistema satisface las necesidades del usuario. Con expertos y usuarios. ¿Cuándo?
- 2 tipos:
 - Objetiva: basada en especificaciones formales
 - Interpretativa: Actividades encaminadas a eliminar los errores de tipo conceptual y de contexto. A veces denominada evaluación.

Cumplir especificaciones del modelo de diseño

- La representación elegida es la adecuada.
- La técnica de razonamiento elegida es la apropiada.
- Reflejo de modelo conceptual en la implementación.
- En el diseño y la implementación se ha pensado en la modularidad.
- La comunicación entre los subsistemas es adecuada.
- El sistema es fácil de mantener y de comprender.

Aspectos de Validación

- ¿Qué se está validando?
- Metodología de validación.
- Criterios de validación.

¿Qué se está validando?

- La comunicación del sistema con otros sistemas (transferencias) es adecuada.
- El interfaz es comprensible para el usuario.
- La explicación del razonamiento del sistema es suficiente.
- Cumple los requisitos de ejecución en tiempo real pedidos.
- El sistema cumple las especificaciones de seguridad.
- Satisfacción y utilidad de los resultados finales e intermedios comparados con: resultados conocidos, prestaciones de un experto o de un modelo algorítmico.

Metodología de validación

- Informal: reuniones.
- Mediante casos de test. Analogía con Turing. ¿Significación y corrección de los casos?
- Pruebas de campo. Actuación en paralelo con experto.
- Validación de subsistemas (muy importante, imprescindible en sistemas mínimamente complejos)
- Análisis de sensibilidad. Sistemas con incertidumbre: cambios provocados por los posibles casos que se puedan dar en la incertidumbre. Pequeños cambios no deberían dar resultados muy dispares.

Criterios de validación

- Cuántos casos de prueba.
- Cómo se generan estos casos de prueba.
- Establecer una proporción entre casos fáciles, medios y difíciles.
- Cómo comparar los resultados con los de un experto.
- Cómo se mide la realización de un experto humano en ese campo.
- Cómo evaluar el sistema cuando distintos expertos opinan distintas cosas.
- Si se generan distintas respuestas cómo se consideran.

Resultados del proceso de validación

- Exactitud u aceptabilidad de soluciones.
 - ¿Cuántas veces acierta?
- Adecuación al problema.
 - ¿Cubre el dominio? ¿responde ante todos los casos reales que se puedan dar?
- Errores:
 - Por comisión: (se responde incorrectamente)
 - Por omisión: (no se responde)

Pasos de Verificación y Validación

- Verificar si el sistema es completo, correcto y consistente.
- Evaluar si el sistema cumple especificaciones del modelo de diseño.
- Diseñar un plan de validación aplicando metodologías apropiadas.
- Valorar en función de criterios de validación. Entre otros los requisitos funcionales definidos en la fase de identificación del problema.