Chatterbot Inteligente para Asesoramiento Jurídico

María Daniela López De Luise [#1], Patricio Carrilero[#2], Andrés Pascal [*1], Rafael Malgor[*2], Claudia Alvarez[&1], Joaquin Diaz[&2], Pablo Pescio[&3], Ben Saad [&4]

[#] *CI2S Lab*

Ciudad Autónoma de Buenos Aires, Argentina

[1] daniela_ldl@ieee.org

[2] patriciocarrilero@gmail.com

[*] *Departamento de Ingeniería en Sistemas de Información, FRCU, UTN*

Concepción del Uruguay, Entre Ríos, Argentina

[1] andrespascal2003@yahoo.com.ar

[2] rafaelmalgor@gmail.com

[&] *Departamento de Ciencia y Tecnología, UADER*

Concepción del Uruguay, Entre Ríos, Argentina

[1] claudialvarez2000@yahoo.com.ar

[2] mjoaquind@gmail.com

[3] pdpescio@yahoo.com.ar

[4] ben.saad@gmail.com

***Abstract*— This paper presents the first results of a functional prototype implementing a linguistic model focused on regulations in Spanish. Its global architecture, the reasoning model, a case-study and short statistics are provided for the prototype named PTAH. It mainly has a conversational robot linked to an Expert System by a module with many intelligent linguistic filters, implementing the reasoning model of an expert. It is focused in bylaws, regulations, jurisprudence and customized background representing entity mission, vision and profile. This structure and model are generic enough to self adapt to any regulatory environment, but as a first step, it was limited to academic field. This way it is possible to limit the slang and data number. The foundations of the linguistic model and the way the architecture implements the key features of the behavior, are also outlined. The cases presented are a few just to show the usability, flexibility and prospectives of this proposal.**

Keywords—Computational Linguistic, Linguistic Reasoning, Natural Language Processing, Text Mining, Chatterbot, Legal Advice, Semantics, Data Mining, Expert Systems

***Resumen*— Este artículo presenta los primeros resultados de un prototipo funcional implementando un modelo lingüístico centrado en la normativa local. Se proporcionan la arquitectura global del modelo de razonamiento, un estudio de casos y algunas estadísticas para el prototipo llamado Ptah. El mismo es principalmente un robot conversacional vinculado a un sistema experto por un módulo con filtros lingüísticos inteligentes, que implementan un modelo de razonamiento experto. Su pericia versa sobre estatutos, reglamentos y jurisprudencia. La estructura y el modelo son capaces de autoadaptarse a cualquier entorno normativo. A pesar de ello, como un primer paso se considera sólo legislación académica. Esto permite trabajar con un argot y volumen de datos más reducido y manejable. También se presentan aquí las bases del modelo lingüístico interno y la forma en que la arquitectura implementa las características esenciales del razonamiento. Asimismo se presentan casos a fin de mostrar la facilidad de uso, flexibilidad y perspectivas de esta propuesta.**

Palabras claves: Lingüística Computacional , Razonamiento Lingüístico, Procesamiento del Lenguaje Natural, Minería de texto, Chatterbot, Asesoramiento Jurídico, Semántica, Minería de datos, Sistemas Expertos

I. Introducción

A principios de los ‘50, Alan Turing propuso el famoso Test de Turing, uno de los principales retos en el campo de la Inteligencia Artificial. Este test intenta definir lo que puede ser la inteligencia proporcionada por un ordenador, y al mismo tiempo, la posibilidad de que las máquinas puedan imitar el razonamiento humano [1].

J. Weizenbaum continuó esa idea pero desde otra perspectiva, construyendo un prototipo llamado Eliza [2]. Pero Eliza no es sólo un programa de ordenador, sino que es uno de los primeros ejemplos de Procesamiento del Lenguaje Natural (NLP), basado en la identificación de patrones predefinidos como base para la comprensión del lenguaje, que constituye uno de los primeros robots conversacionales (chatter bots o chat bots). Años más tarde, el Dr. Colby crea Parry [6], otro chat bot que imita el comportamiento de un paciente psiquiátrico que sufre de paranoia. Éste puede generar diálogos de acuerdo a diferentes tipos de paranoia. Las pruebas muestran que al menos un conjunto de psiquiatras no fue capaz de distinguir entre el ordenador y el paciente humano. [3]

Sobre la base de ELIZA [2], Richard Wallace desarrolló más tarde un proyecto llamado Alice (1995) [5]. Como uno de los derivados de este proyecto, surgió el AIML (Artificial Intelligence Mark-up Language), lenguaje similar a XML que tiene una marca general (etiqueta) llamada Categoría, que representa la unidad elemental de conocimiento. Cada categoría de conocimiento tiene dos componentes: Patrones y Modelos. El patrón es una cadena de caracteres que representa una sentencia dentro del diálogo, y el modelo representa la respuesta al patrón que está siendo activado [7].

El proyecto PTAH tiene como interfaz usuario un chatterbot, pero también funciona como un filtro inteligente,

ya que su argot está relacionado con los reglamentos y los instrumentos jurídicos del ámbito académico.

Para que el sistema sea preciso, es importante que sea capaz de superar problemas como la ambigüedad, polisemia, la anáfora, etc., lo que se logra al identificar el contexto adecuado de la consulta a resolver. La mayoría de las propuestas actuales se encuadran en enfoques colectivamente conocidos como Procesamiento del Lenguaje Natural (NLP). [4] [8]

La propuesta típicamente involucra uno o más de los siguientes niveles de análisis: fonológico, morfológico, sintáctico, semántico y pragmático. Pero las propuestas rara vez cubren todos ellos al mismo tiempo. Este enfoque por capas, es útil para poder descomponer el problema en partes más simples y de este modo hacerlo más sencillo. Muchas veces los autores proponen estrategias basadas en grandes diccionarios y cierto pre-procesamiento que puede llegar a ser costoso y/o complejo. Por lo general, se trabaja en base a un corpus. Muchas propuestas requieren (en mayor o menor grado) cierta interacción humana [9].

En cuanto a la capa semántica (SFW), también hay muchas propuestas que complementan las iniciativas anteriores. Entre otras cabe mencionar WebODE [10] [11] [12], y la ingeniería ontológica, que permiten el desarrollo de sitios Web para manejar cierto tipo de conocimiento de forma casi automática. Otro SFW importante es ContentWeb, una plataforma para la integración ontológica con WebODE, que permite al usuario interactuar utilizando lenguaje natural pero con una jerga bien limitada [13]. Ese ambiente interactúa también con OntoTag (implementado con RDF / S y XML), OntoConsult (interface de lenguaje natural basado en ontologías) y OntoAdvice (un sistema de recuperación de información basado en ontologías). En este contexto, cada palabra recibe un identificador llamado URI (Uniform Resource Identifier). También se define un URI para cada elemento morfosintáctico. Existen también entornos que sirven para manejar la morfosintaxis de los textos. Por ejemplo XTAG desarrolla una muy buena gramática inglesa [15] basándose en un modelo de lexicalización llamado Árbol Gramatical (TAG), que consiste en una estructura de árbol resultante del procesamiento de la sintaxis. Incluye un analizador (módulo que divide el texto en palabras), una interfaz X-windows y un analizador morfológico.

Entre otros múltiples ejemplos, es importante mencionar una herramienta [16] que realiza el análisis morfológico y sintáctico con segmentación desambiguada (divide el texto en segmentos según su coherencia), desambiguación de símbolos especiales (utilizado para sonidos no relacionados con palabras) y corrección de errores (cuando las palabras son mal interpretadas).

El presente trabajo se encuadra en el problema de desarrollos de chatbots y lo ensambla con un modelo lingüístico de razonamiento en Español, originando una visión alternativa del procesamiento textual basado en características gramaticales. Para ello, introduce en el chatbot una tecnología adicional relacionada con los Sistemas Expertos y la distancia semántica en un área de conocimiento reducido [25]. Esta tecnología se probó con éxito en otros problemas que requieren de gestión automática del contexto como método natural de trabajo [26]. Específicamente ha sido probado en el procesamiento automático de los diálogos en lengua española [27]. Este trabajo se emplea en morfosintaxis pero queda pendiente de aplicación para la detección de perfiles y otros mecanismos derivados de datos como la extracción automática de características a partir de datos históricos, que complementaría los hallazgos y derivaría en un mejor modelo del contexto que el que se presenta en este trabajo.

El marco puede ser pensado como un razonamiento lingüístico por capas de la siguiente manera:

1. Primer paso: Filtrar oraciones usando el modelo lingüístico. Clasificar el tema de consulta automáticamente utilizando estructuras de palabras clave (en lugar de semánticas explicitadas por etiquetas, estructuras o diccionarios). Estas estructuras son previamente aprendidas y representan ciertas características morfosintácticas de una secuencia específica de palabras. Son manipuladas por un Sistema Experto basado en reglas.
2. Segundo paso: permite una asociación semántica a nivel de palabras, utilizando una distancia no métrica basada en relaciones predefinidas.

Las secciones que siguen son: descripción de propuesta de distancia semántica (sección II), presentación de un conjunto de distancias métricas que se pueden utilizar opcionalmente en lugar de la opción definida en el prototipo (sección III), esquema del proyecto y del modelo lógico (sección IV), conclusiones y trabajo futuro (sección V).

II. Medidas de Similitud y Semánticas

Las búsquedas utilizando similitud tienen un gran número de aplicaciones tales como el reconocimiento de imágenes y de sonidos, la compresión y las búsquedas en textos en áreas de sistemas bioinspirados, en inteligencia computacional, minería de datos, etc [17]. En todos los casos existe la misma característica: se buscan similitudes utilizando cierta función de distancia o similitud predefinida para ese caso. El modelo más utilizado es el basado en espacios métricos.

Un espacio métrico se define como un par **(U, d)** siendo **U** el universo de objetos y **d: U x U** $\rightarrow$ $\mathbf{R}^+$ una función de distancia definida para evaluar la similitud entre los elementos de **U**. Esto significa que la distancia más corta se obtiene para los objetos más cercanos. Esta función **d** también cuenta con las propiedades típicas de una distancia métrica:

Positividad:

$$x, y \quad U, d(x, y) \geq 0 \quad , \tag{1}$$

Simetri

Simetría:

$$x, y \quad U, d(x, y) = d(y, x) \quad , \tag{2}$$

Reflexividad:

$$x \quad U, d(x, x) = 0 \quad , \tag{3}$$

Desigualdad triangular:

$$x, y, z \quad U, d(x, y) \leq d(x, z) + d(z, y) \quad , \tag{4}$$

La base de datos de trabajo, es un subconjunto finito denotado como ***X***, ampliamente incluido en ***U*** con cardinalidad ***n***. En este modelo, una consulta típica implica recuperar los objetos similares a uno dado utilizando búsquedas para ciertos rangos. Sea ***(*q, r)** una consulta donde **q** $\in$**U** y ***r*** es un radio de tolerancia al error, una

búsqueda por rango consiste en recuperar todos los objetos ***s*** en la base de datos que tienen una distancia a ***q***, menor o igual que ***r***. Esto es lo que se expresa en (Ec. 5).

$$(q, r) = \{x \in X / d(q, x) \leq r\}\ , \qquad (5)$$

Una búsqueda por rango devuelve los resultados con una complejidad de orden O(n) si se recorre exhaustivamente la Base de Datos (DB). Para evitar ésto, es posible procesar previamente la DB con un algoritmo que construye un índice, ahorrando tiempo de búsqueda durante el cálculo. Un algoritmo de indexación es eficiente si puede responder una consulta usando el concepto de similitud y realizando una mínima cantidad de cálculos de distancia, típicamente sub-lineal sobre el número de elementos en la Base de Datos (esto es, su orden es menor a O(n)) [20] [21] [23]. Este proyecto aplica este criterio para mejorar la distancia semántica y proporcionar un algoritmo económico.

III. Métricas de Distancia

El análisis de métricas anterior sirve como introducción para entender cómo una buena distancia debe comportarse en cuanto a la eficiencia. Teniendo esto en cuenta, es importante notar que las distancias dependen en gran medida de la cantidad y calidad de las características que las constituyen. Entre las métricas más famosas están las distancias Euclidiana y Manhattan [17]. Pero existen muchas otras que están bajo consideración y evaluación como parte de este proyecto. Se describen a continuación sólo para demostrar el alcance del proyecto global. Esta parte de la investigación está destinada para la evaluación de la flexibilidad del modelo y también para que quede claro cómo se puede mejorar.

A. Overlap Metric (OM)

Cuando todos los atributos son nominales [24], la distancia métrica más simple es la Superposición Métrica (OM) definida como:

$$d(x, y) = \sum_{i=1}^{n} \delta\left(a_i(x), a_i(y)\right)\ , \qquad (6)$$

Donde:
- n cantidad de atributos,
- $a_i(x)$ y $a_i(y)$ valores del atributo i-ésimo a_i para las instancias x e y respectivamente,
- $\delta(a_i(x), a_i(y))$ es 0 cuando $a_i(x) = a_i(y)$ y 1 en otro caso.

Usos: OM es ampliamente utilizado para el aprendizaje basado en casos y el aprendizaje con ponderación local.

Debilidad: es básico, poco preciso para evaluar la distancia entre cada par de casos, ya que no hace uso de ninguna información adicional de los datos que pudiera servir para mejorar la función.

B. Value Difference Metric (VDM)

Es una versión simplificada de las anteriores [24] utilizando ponderaciones:

$$d(x, y) = \sum_{i=1}^{n} \sum_{c=1}^{C} \left| P\left(c \quad a_i(x)\right) - P\left(c \quad a_i(y)\right) \right| \ , (7)$$

Donde:
- C cantidad de clases
- $P(c|a_i(x))$ la probabilidad condicional para que x sea de clase C, dado un atributo $a_i = a_i(x)$
- $P(c|a_i(y))$ la probabilidad condicional para x sea de clase C, dado un atributo $a_i = a_i(y)$

VDM asume que dos atributos están más cerca cuando sus clasificaciones son similares (la evidencia muestra que es más precisa que la OM).

C. Métricas SFM

Distancia definida como [24]:

$$d(x, y) = \sum_{c=1}^{C} \left| P\left(c \quad x\right) - P\left(c \quad y\right) \right|\ , \qquad (8)$$

Aquí las probabilidades P(c|x) y P(c|y) sólo pueden estimarse utilizando Bayes, ya que otros estimadores pueden requerir una mayor complejidad de evaluación y tomar mucho más tiempo durante la búsqueda. SFM minimiza la expectativa entre la diferencia del error finito y el error asintótico.

D. Minimum Risk Metric (MRM)

MRM [24] minimiza el riesgo de errores de clasificación. La ecuación siguiente define su cálculo:

$$d(x, y) = \sum_{c=1}^{C} P\left(c \quad x\right)\left(1 - P\left(c \quad y\right)\right)\ , \qquad (9)$$

E. Métricas usando entropía

Esta métrica se basa en teoría de la información [24]. Considera la distancia entre instancias como la complejidad requerida para convertir un objeto en otro. La ventaja de esta métrica es que proporciona una aproximación coherente para atributos simbólicos tan buena como para los valores reales y los valores que fueron omitidos.

F. Metric Data-driven

Es útil para datos categóricos [24], donde la diferencia entre los datos se basa en la frecuencia de las categorías o alguna combinación de las categorías que pueden compartir.

IV. El Modelo y Ptah

El modelo implementado en PTAH tiene los módulos descritos en la fig. 1.

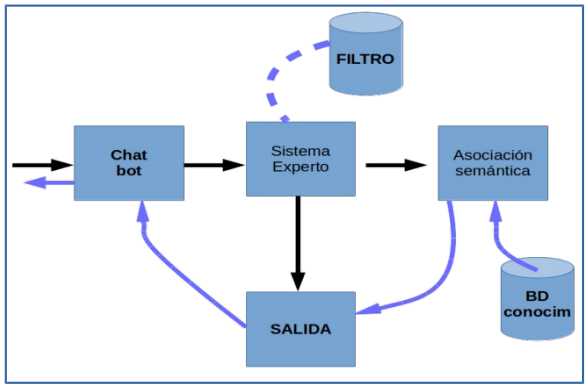
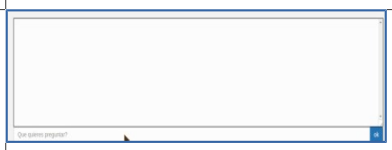
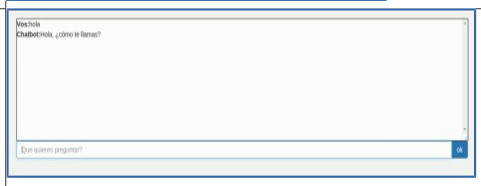
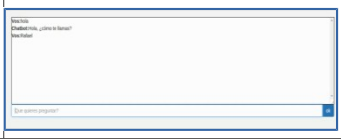
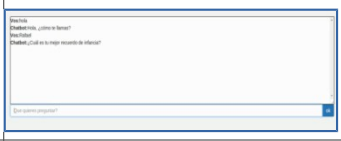


Fig.1.Arquitectura global de Ptah

A. *Chatterbot*

Es el módulo de entrada. Es un robot conversacional con patrones codificados en Python. La Tabla I detalla una secuencia de capturas de pantalla usando la interface durante el desarrollo de un caso ejemplo.

TABLA I
INTERFACE DE DATOS DE ENTRADA A TRAVÉS DEL CHATTERBOT

Actividad	**Pantalla**
inicio	
bienvenida	
identificación	
consulta	
respuesta	

B. *El Sistema Experto (SE)*

Implementado en Python como un conjunto de módulos que identifica los temas de consulta. Comunica a los siguientes módulos los temas para su pre-procesamiento. La calidad de estos módulos no sólo determina la morfosintaxis sino también se utiliza para el procesamiento temprano de la semántica.

El SE tiene un conjunto de reglas que describen los casos de uso de interés para los usuarios. Una breve lista de ellos se describe en la tabla II.

TABLA II.
DETALLE DE LOS CASOS Y SU RELACIÓN CON LAS REGLAS DEL SISTEMA EXPERTO

ID caso	Caso	Regla
1.	¿Qué diferencia hay entre ser docente interino o concursado (ordinario)?	diferencia {interino+concursado} {cuándo|condiciones} + interino {cuándo|condiciones} + {concursado|ordinario}
2.	¿Cuándo me otorgaron el cargo que gané en el concurso de X materia? ¿Cuál es el número de resolución?	{cuándo+otorgaron} + materia X {qué|número} + resolución + designación + materia X resolución + {concurso|para} + materia X
3.	¿Cuál son las composiciones de los Consejos Departamental, Directivo y Superior?	[Consejo] + {Departamental| Directivo| Superior}+ composiciones {compone| miembros} + consejo + {departamental|directivo| superior}

C. *Asociación Semántica*

Implementada como un conjunto de procedimientos para perfilar las asociaciones simples y complejas entre los elementos de la consulta. La misma genera una cadena de caracteres que resume una matriz optimizada de números que representan semánticamente las palabras involucradas. Independiente del caso específico, el razonamiento se representa por el siguiente algoritmo:

1.Find Meta data(Stemming (data))
2.Find Binary vector(Meta data)
3.Retrieve binary vector using:
3.Bias A: restriction for no divergence
3.1.Binary vector from every bylaw
3.2.relative frequency for every word (w) in the bylaw → freq(w)
3.3 MIN(w)= argMIN{ freq(w)}
3.4.MAX(w)= argMIN{ freq(w)}
3.5.find weighting for every w: p(w)=1/freq(w)
*3.6.p(w)=1/(MAX(w)*1.05)*
3.Bias B: relevance in the current case
3.1.Binary vector(ID-case)
3.2.NUM(w)= number of words in ID-case
3.3.Let p'(w_i) = 1/NUM(w_i) for every w_i
3.Bias C: relevance in the current context
3.1.Binary vector(query(data))
3.2.For every w_i(Meta data) and every ID-case:
3.2.1.IF w_i(Meta data) AND w_i(ID-case) AND match (ID-case) scoring(ID-case)+= p'(w)
*3.2.2.IF w_i(Meta data) AND w_i(ID-case) ~match(ID-case) scoring(ID-case) += p'(w) * 0.95*
3.2.2.ELSE scoring(ID-case) - =p'(w) / there is no w_i(ID-case) in knowledgeDB */*
3.3.select argMAX{scoring(ID-case)}
3.3.1.IF number-of(ID-case) >1 THEN ID-caseBEST=select argMAX{freqH(ID-case)}
3.Bias D: hit precision in the DB
3.1.search KnowledgeDB (binary vector(ID-caseBEST))
3.2. IF(hit (ID-caseBEST)) -> scoring += p(w)
*3.3. IF(~hit(ID-caseBEST) -> scoring += p(w) * 0.95*
3.4. ELSE / there is no w_i(ID-case) in knowledgeDB */*
3.4.1.IF hit(w_i(ID-case))-> scoring - = p (w)
3.4.2.IF ~hit (w_i(ID-case))-> scoring - = p' (w)
*3.5.output (select * from KnowledgeDB where argMAX {scoring})*

En el algoritmo, **freqH** representa la frecuencia de uso anterior, compilada durante toda la historia del modelo.

D. *KnowledgeDB*

Se trata de un Sistema de Gestión de Base de Datos PostgreSQL, con el DB presenta en la Fig. 2.

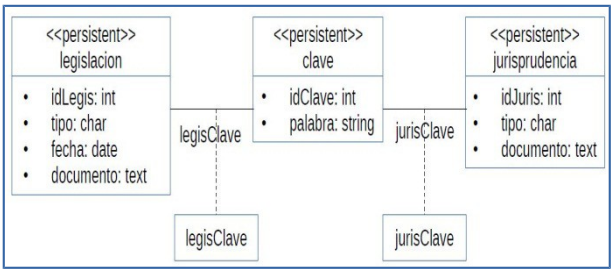


Fig. 2. ERD de la DB conectado al ES

Los accesos se realizan a través de un conjunto de matrices binarias en el módulo de relaciones semánticas. Es importante la función de distancia seleccionada para recuperar la información apropiada, ya que permite direccionar un contenido con más precisión. La Base de Datos se llena con información textual, pero se espera que mejore la carga de datos utilizando también un ICR para incluir los documentos no textuales.

V. Estudio de Caso

En esta sección se muestra cómo el sistema procesa una consulta, los eventos en el proceso de razonamiento del modelo y el tipo de resultados que puede responder.

Paso 1: hacer una consulta

El estudiante consulta la siguiente frase "¿cuando soy alumno regular?"

Paso 2: razonamiento lingüístico

El chatbot reacciona activando ciertas reglas del SE que forman parte de su base de conocimiento, con el fin de definir el tema del problema. En este ejemplo, se detectan cuatro identificadores de temas (IDs) para la consulta. Ellos son el 1, 12, 24 y 39 que se describen abajo. Cada ID (identificador de tipo de consulta o tema) está relacionado con un patrón basado en cómo las palabras se relacionan entre sí. Por el momento, no hay ninguna otra consideración durante el procesamiento de las relaciones de las palabras.

A continuación se muestran los cuatro patrones que corresponden a la consulta:

Consulta 1:{ requisitos | necesita | pautas | cuáles + puntos | cuándo | qué} + alum + regular
Consulta 12: {cuándo | condiciones} + interino
Consulta 24: {quienes + pedir | condiciones | facilidades requisitos | cuándo} + mesas
Consulta 39: {docente (s) | alumno (s)} + intercambio (s)}

Con:
{|} Denota alternativa entre palabras
[ ] Denota que la palabra no es requerida. Básicamente palabras que pueden o no estar en la frase
(xx) sufijos de la palabra que no son obligatorios
+ Palabra obligatoria (que no implica ningún tipo de orden)

En el ejemplo, la consulta será considerada como un tema relacionado con el ID 1 si y sólo si se logran todos los siguientes requisitos:
a) al menos comprende una de las palabras siguientes: *requisitos*, *necesita*, *pautas*, *cuáles puntos*, *cuándo*, *qué*
b) en algún lugar existe la palabra *alum*
c) en algún lugar existe la palabra *regular*

Si alguno de ellos no está, entonces la consulta no se relaciona con la pregunta 1 y por lo tanto la ID 1 no corresponde. En el ejemplo la consulta coincide y como consecuencia ID 1 es uno de los candidatos a ser el tema a procesar en la DB. Este resultado se procesa luego en el Paso 3 que se describe a continuación.

Paso 3: asociaciones semánticas

Después del filtrado inteligente inicial por el SE, es el momento para que las asociaciones semánticas encuentren cualquier dato relacionado con el tema actual.

Paso 4: Salida

Después de acceder a la Base de Datos hay sólo un caso identificado en las matrices binarias como índice de contenidos que se relacionan. Los datos se acceden usando dicho índice, se recuperan y se responden a través del chattbot.

TABLA III
PRESENTA LOS RESULTADOS DE ALGUNAS OTRAS CONSULTAS.

ID-query	Query	ID-CASE	Hit	Precision	Recall
1	Cuando soy alumno regular?	1,12,24, 39	yes	0.25	1
2	Cuál es el régimen de correlativas	6	no	0	0
3	Cuántas faltas puedo hacer en Algebra	5,6,7,32, 38	no	0	0
4	Qué pasa si falto a un final?	1,2,5,13, 18, 45,46	yes	0.14	1

De la tabla se puede decir que:
PRECISION (Precisión, cantidad de casos recuperados / casos recuperados relevantes) es bastante bajo. La razón de ello es el tipo de función de distancia entre palabras (euclidiano). También influye la poca cantidad de documentos en la base de datos (actualmente 17), pero aún con escasos documentos el número de ID-casos es mayor que 0 (esa información aún debe pasar por los pasos de procesamiento). En el caso de las consultas 2 y 3, el número de casos en BD recuperados es 0 (pues no hay ningún documento en la actual Base de Datos con ese tema).
En cuanto al RECALL (recuperación de casos, cantidad de casos relevantes recuperados / número de casos relevantes en la DB), es del 50% para el total de casos y de un 100% para los casos con hit. Queda pendiente aún una evaluación estadística con un número mayor de consultas.

VI. Conclusiones y Trabajo Futuro

Este artículo presenta un modelo de razonamiento lingüístico para diálogos compatible con el enfoque semántico indirecto típico de los modelos basados en morfosintaxis, pero mejorado por un heurístico dirigido por datos. El prototipo PTAH es una implementación de dicho modelo. Extiende el procesamiento de un chatterbot tradicional usando nuevas capas de abstracción que no encajan en las estrategias conocidas al momento en el área de NLP. En estas capas se distribuyen filtros en un SE basado en reglas que operan sobre un conjunto de criterios explícitos en su estructura:

Sesgo A: restricción de no divergencia
Sesgo B: relevancia en el caso actual
Sesgo C: relevancia en el contexto actual
Sesgo D: precisión en cada caso

Se ha presentado la arquitectura global y un estudio de caso. Es importante tener en cuenta que en esta estrategia no se requiere etiquetados, diccionarios o corpus entrenados. De los resultados preliminares aquí presentados se puede afirmar que las métricas PRECISION y RECALL son bastante buenas a pesar de que la métrica de distancia es pobre y se puede mejorar explorando funciones de distancia más complejas. Entre las tareas pendientes quedan:

- Insertar y considerar los diccionarios y los datos históricos para la reutilización de las consultas
- Ajuste automático de reglas en el ES, también las reglas pueden ser aprendidas probabilísticamente de la historia.
- Evaluar otras métricas de distancia que puedan evidenciar relaciones lingüísticas entre palabras. Esto mejoraría la respuesta ante situaciones nuevas y haría que el sistema sea más flexible.
- Evaluar la carga, precisión y recuperación de casos con alto número de consultas
- Evaluar los mismos parámetros utilizando las distancias de la sección III.
- Implementar un nuevo módulo ICR y carga automática de la Base de Datos.
- Mejorar la interface usando un sintetizador y un sistema de reconocimiento de voz. Esto puede hacer que la interacción sea más amigable.
- Extender los casos de uso a otros temas y mejorar el chatterbot para ser menos sensible a dialectos.

REFERENCIAS

[1] Turing, Alan (1950), "Computing Machinery and Intelligence", Mind 59: 433-60.

[2] J. Weizenbaum. "ELIZA- A Computer Program for the Study of Natural Language Communication between Man and Machine", Communications of Association for Computing Machinery 9 (1966): 36-45.

[3] J.Weizenbaum. "Computer power and human reason". San Francisco, CA. W.H. Freeman, 1976.

[4] T. Winograd. "Procedures as a Representation for Data in a Computer Program for Understanding Natural Language". Cognitive Psychology Vol.3 Nº 1, 1972.

[5] ALICEBOT, alicebot.blogspot.com/

[6] K.M Colby, F.D. Hilf, S. Weber, J. Kraemer. "Turing-Like Indistinguishability Tests for the Validation of a Computer Simulation of Paranoid Processes". A.I., 3 (1972) pp199-222.

[7] R.Wallace. "The Elements of AIML Style". ALICE AI FOUNDATION, 2003.

[8] Manning C., Schütze H. "Foundations of Statistical Natural Language Processing". MIT Press.1999.

[9] M. Mauldin. "Chatterbots, TinyMuds and The Turing Test: Entering The Loebner Prize Competition". AAAI-94, 1994.

[10] Corcho O., López Cima A., Gómez Pérez A. "A Platform for the Development of Semantic Web Portals". ICWE'06. Palo Alto. California. USA. ACM. 2006.

[11] Corcho O., Fernández-López M., Gómez-Pérez A. Vicente O. "WebODE: an integrated workbench for ontology representation, reasoning and exchange". Knowledge Engineering and Knowledge Management Proceedings (2002). Volume: 2473, Pages: 138-153

[12] http://webode.dia.fi.upm.es/WebODEWeb/index.html

[13] Aguado de Cea G., Álvarez de Mon y Rego I., Pareja Lora A. "Primeras aproximaciones a la anotación lingüístico-ontológica de documentos de la Web Semántica: Onto Tag". Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial. No. 17. pp. 37 - 49. 2002.

[14] Aguado de Cea G., Álvarez de Mon y Rego I., Pareja Lora A., Plaza ArtecheR."RFD(S)/XML LINGUISTIC ANNOTATION OF SEMANTIC WEB PAGES". International Conference on Computational Linguistics. Proceedings of the 2nd workshop on NLP and XML - Volume 17.pp 1 - 8. 2002.

[15] Paroubek P., Schabes Y., Joshi A. K. "XTAG - A Graphical Workbench for Developing Tree-Adjoining Grammars". Third Conference on Applied Natural Language Processing, Trento (Italy). 1992

[16] Prószéky G., Naszódi M., Kis B. "Recognition Assistance: Treating Errors in Texts Acquired from Various Recognition Processes". International Conference on Computational Linguistics. Proceedings of the 19th international conference on Computational linguistics - Volume 2.pp 1 - 5. 2002.

[17] E. Chavez, G. Navarro, R. Baeza-Yates, and J.L. Marroquın. Searching in metric spaces. ACM Computing Surveys, 33(3):273–321, September 2001.

[18] J. Vitter. External memory algorithms and data structures: Dealing with massive data. ACM Computing Surveys, 33(2):209–271, 2001

[19] D. Clark and I. Munro. Efficient suffix tree on secondary storage. In Proc. 7Th ACM-SIAM Symposium on Discrete Algorithms, pages 383–391, 1996.

[20] R. Baeza-Yates, W. Cunto, U. Manber, and S. Wu. Proximity matching using fixed-queries trees. In Proc. 5th Combinatorial Pattern Matching (CPM'94), LNCS 807, pages 198–212, 1994.

[21] CHAVEZ, E., MARROQUiN, J., AND NAVARRO, G. 2001a. Fixed queries array: A fast and economical data structure for proximity searching. Multimed. Tools Appl. 14, 2 (June), 113–135. (Expanded version of Overcoming the curse of dimensionality. In European Workshop on Content-Based Multimedia Indexing, pages 57–64, Toulouse, France, October 1999).

[22] BUGNION, E., FEI, S., ROOS, T., WIDMAYER, P., AND WIDMER, F. 1993. A spatial index for approximate multiple string matching. In Proceedings of the 1st South American Workshop on String Processing (WSP'93) (Belo Horizonte, Brazil), R. Baeza-Yates and N. Ziviani, Eds. 43–53.

[23] CHAVEZ, E. AND NAVARRO, G. 2000. An effective clustering algorithm to index high dimensional spaces. In Proccedings String Precessing and Information Retrieval (SPIRE 2000) (A. Coruna, Spain), 75-86.

[24] Chaoqun Li and Hongwei Li. A Survey of Distance Metrics for Nominal. Attributes.JOURNAL OF SOFTWARE, VOL. 5, NO. 11. Department of Mathematics, China University of Geosciences, Wuhan, Hubei, China 430074. Email: fchqli, hwlig@cug.edu.cn . 2010.

[25] "Morphosyntactic Linguistic Wavelets for Knowledge Management". Libro "Intelligent Systems", ISBN 979-953-307-593-7. InTech OPEN BOOK. 2011.

[26] "Language Modeling with Morphosyntactic Linguistic Wavelets ". D. López De Luise, D. Hisgenr. "Automatic content extraction on the web with intelligent algorithms". CIDM. 2013.

[27] "Modeling dialogs with Linguistic Wavelets". D. López De Luise,D. Hisgen, A. Cabrer, M. Morales Rins. IADIS Theory and Practice in Modern Computing 2012 (TPMC 2012). Lisboa, Portugal. 19 a 21 de Julio del año 2012.

[28] Dissimilarity learning for nominal data. Victor Cheng, Chun-Hung Li , James T. Kwok , Chi-Kwong Li. Journal of Pattern Recognition. Elsevier. 2003.