



**UNIVERSIDAD TECNOLÓGICA DE LA MIXTECA**

**“DESARROLLO E IMPLEMENTACIÓN DE UN PÁRSER  
SEMÁNTICO PARA EL MÓDULO GOLEM-UNIVERSUM”**

**TESIS**

**PARA OBTENER EL TÍTULO DE  
INGENIERO EN COMPUTACIÓN**

**PRESENTA**

**ALDO ERNESTO FABIÁN AGUILAR**

**DIRECTORES DE TESIS**

**DR. MANUEL HERNÁNDEZ GUTIÉRREZ**

**DR. IVAN V. MEZA RUÍZ**

**HUAJUAPAN DE LEÓN, OAXACA. ABRIL DE 2011.**



*A mis padres, Máximo y Elizabeth,  
por ser mi motivación e inspiración  
para alcanzar mis objetivos y cumplir  
mis metas. Los quiero mucho.*



## Agradecimientos

A mis padres agradezco sinceramente su presencia en cada uno de los días de mi vida. Por todas las enseñanzas, consejos y regaños, por compartir tanto mis aciertos como mis errores, pero sobre todo por ese cariño incondicional que siempre me han dado. Gracias papá. Gracias mamá.

A mis hermanos Alina y Alonso por las experiencias vividas y las que aún nos faltan, por el cariño y el apoyo que me han dado a través del tiempo, gracias.

A mis abuelita Ino y a mi abuelito Martín gracias por estar siempre conmigo a pesar de la distancia, por estar al pendiente de mi y por hacerme saber lo importante que soy para ustedes. Los quiero mucho.

A mis tíos les agradezco sinceramente la confianza que me han tenido así también el apoyo incondicional me han dado. Son para mi un gran ejemplo, tanto como profesionistas como seres humanos. Los admiro, los respeto pero sobre todo los quiero.

Al Dr. Iván Meza Ruíz le agradezco el aceptar dirigir este trabajo porque sin su guía este trabajo no hubiera sido posible. Gracias por compartir sus conocimientos y por el compromiso mostrado. Muchas gracias.

Al Dr. Manuel Hernández Gutiérrez por su apoyo para la realización de este trabajo así como sus comentarios y observaciones.

A mis sinodales Dr. José Aníbal Arias Aguilar, Dr. Santiago Omar Caballero Morales y Dr. Felipe de Jesús Trujillo Romero por la disposición que tuvieron para la revisión de este trabajo y así también por sus comentarios y observaciones que ayudaron a mejorarlo.

Al Dr. Luis Pineda agradezco la oportunidad de pertenecer a su equipo de trabajo, primero en mis estancias profesionales y después para el desarrollo de este trabajo.

A las personas del DCC les estoy agradecido por las experiencias vividas, la oportunidad de conocerlos y el apoyo que me mostraron durante mi estancia.

A los proyectos CONACYT 81965 y PAPPIT-UNAM IN-115710 por proporcionar los recursos y la infraestructura necesarios para poder concluir de manera exitosa este trabajo.

A todas y cada una de las personas que se vieron involucradas, directa o indirectamente, con la realización de este trabajo, gracias.

# Índice general

Índice general	VII
Índice de figuras	XI
Índice de cuadros	XIII
<b>1. Introducción</b>	<b>1</b>
1.1. Sistemas de diálogo . . . . .	3
1.2. Procesamiento semántico . . . . .	9
1.3. Objetivos . . . . .	12
1.3.1. Objetivos específicos . . . . .	12
1.4. Estructura de la tesis . . . . .	13
<b>2. Trabajo previo</b>	<b>15</b>
2.1. Estado del arte en procesamiento semántico . . . . .	16
2.2. Proyecto Golem-UNIVERSUM . . . . .	24
2.2.1. Estructura del sistema . . . . .	25
2.2.2. Procesamiento semántico . . . . .	28
2.2.3. Representaciones semánticas . . . . .	30

---

<b>3. Metodología propuesta</b>	<b>35</b>
3.1. Corpus Golem-UNIVERSUM . . . . .	36
3.2. Lógica de Markov . . . . .	39
3.2.1. Redes Lógicas de Markov . . . . .	40
3.2.2. Lógica de Markov como modelo gráfico . . . . .	42
3.3. Métricas de evaluación . . . . .	44
<b>4. Procesamiento semántico sin contexto</b>	<b>49</b>
4.1. Experimentos con transcripciones manuales . . . . .	50
4.1.1. Baseline . . . . .	50
4.1.2. Experimento parametrizador . . . . .	52
4.1.3. Modelos extendidos . . . . .	55
4.1.4. Evaluación . . . . .	61
4.2. Experimentos con transcripciones automáticas . . . . .	64
4.2.1. Baseline . . . . .	64
4.2.2. Experimentos parametrizadores . . . . .	65
4.2.3. Evaluación . . . . .	68
4.3. Marco de los experimentos . . . . .	70
4.4. Discusión . . . . .	71
<b>5. Procesamiento semántico con contexto</b>	<b>73</b>
5.1. Experimentos con transcripciones manuales . . . . .	74
5.1.1. Baseline . . . . .	74
5.1.2. Experimentos parametrizadores . . . . .	75
5.1.3. Evaluación . . . . .	80
5.2. Experimentos con transcripciones automáticas . . . . .	83



ÍNDICE GENERAL	IX
5.2.1. Baseline . . . . .	84
5.2.2. Experimentos parametrizadores . . . . .	84
5.2.3. Evaluación . . . . .	86
5.3. Discusión . . . . .	90
<b>6. Conclusiones</b>	<b>93</b>
6.1. Conclusiones . . . . .	94
6.2. Contribuciones . . . . .	95
6.3. Trabajo a futuro . . . . .	97
<b>A. Resultados de los experimentos con modelos extendidos</b>	<b>99</b>
<b>B. Modelos en Lógica de Markov</b>	<b>103</b>
<b>Bibliografía</b>	<b>109</b>



# Índice de figuras

1.1. Arquitectura de un sistema de diálogo. . . . .	4
1.2. Módulos de la etapa de entendimiento del lenguaje. . . . .	11
2.1. Ejemplo de variables ocultas y variables observadas. . . . .	17
2.2. Ejemplo de etiquetado utilizando HMM's. . . . .	19
2.3. Grafo asociado a una tarea de clasificación. . . . .	21
2.4. Grafo relacionado a un CRF para una tarea de etiquetado. . . . .	23
2.5. Cartas utilizadas en el juego "Adivina la carta". . . . .	25
2.6. Arquitectura del sistema del módulo Golem-UNIVERSUM. . . . .	27
2.7. Arquitectura cognitiva del sistema del juego "Adivina la carta". . . . .	32
3.1. Grafo de factores para un fórmula. . . . .	43
3.2. Grafo de factores para un posible mundo. . . . .	43
4.1. Esquema de validación cruzada. . . . .	53
4.2. Curva de aprendizaje para el modelo base. . . . .	54
4.3. Modelo extendido por su estructura. . . . .	56
4.4. Modelo extendido agregando información. . . . .	57

---

4.5. Curva de aprendizaje para el modelo bigramas. . . . .	60
4.6. Curva de aprendizaje para el modelo etiquetado. . . . .	61
4.7. Resultados de la evaluación con transcripciones manuales. . . . .	62
4.8. Resultados de la evaluación con transcripciones automáticas. . . . .	63
4.9. Curva de aprendizaje para el modelo base. . . . .	66
4.10. Curva de aprendizaje para el modelo bigramas. . . . .	67
4.11. Curva de aprendizaje para el modelo etiquetado. . . . .	68
4.12. Resultados de la evaluación con transcripciones automáticas. . . . .	69
5.1. Modelo extendido agregando información del contexto. . . . .	77
5.2. Curva de aprendizaje del modelo base. . . . .	78
5.3. Curva de aprendizaje del modelo bigramas. . . . .	79
5.4. Curva de aprendizaje del modelo etiquetado. . . . .	80
5.5. Resultados de la evaluación con transcripciones manuales. . . . .	82
5.6. Resultados de la evaluación con transcripciones automáticas. . . . .	83
5.7. Curva de aprendizaje del modelo base. . . . .	86
5.8. Curva de aprendizaje del modelo bigramas. . . . .	87
5.9. Curva de aprendizaje del modelo etiquetado. . . . .	88
5.10. Resultados de la evaluación con transcripciones automáticas. . . . .	89

# Índice de cuadros

2.1. Expectativas, patrones y palabras claves. . . . .	29
2.2. Extracto de un diálogo entre el usuario y el sistema. . . . .	31
3.1. Ejemplos del corpus Golem-UNIVERSUM. . . . .	37
3.2. Transcripciones por representación semántica para el entrenamiento	38
3.3. Distribución de las expectativas en el corpus . . . . .	38
3.4. Transcripciones por representación semántica para la evaluación . .	39
4.1. Evaluación del pársers con transcripciones manuales. . . . .	52
4.2. Valores del desempeño del modelo base. . . . .	55
4.3. Ejemplos etiquetados del corpus Golem-UNIVERSUM. . . . .	58
4.4. Comparación del desempeño de los modelos. . . . .	60
4.5. Evaluación de los modelos con transcripciones manuales. . . . .	62
4.6. Evaluación de los modelos con transcripciones automáticas . . . . .	63
4.7. Evaluación del pársers con transcripciones automáticas. . . . .	65
4.8. Comparación del desempeño de los modelos. . . . .	67
4.9. Resultados en el aprendizaje de los modelos. . . . .	70

5.1. Evaluación del pársers con transcripciones manuales y expectativas. . . . .	75
5.2. Comparación del desempeño de los modelos con expectativas. . . . .	80
5.3. Evaluación de los modelos con transcripciones manuales. . . . .	81
5.4. Evaluación de los modelos con transcripciones automáticas. . . . .	82
5.5. Evaluación del pársers con transcripciones automáticas. . . . .	84
5.6. Comparación del desempeño de los modelos con expectativas. . . . .	87
5.7. Comparación de las evaluaciones. . . . .	89
6.1. Desempeño del pársers semántico del sistema Golem-UNIVERSUM. . . . .	95
6.2. Comparación del desempeño de los modelos. . . . .	96
6.3. Experimentos realizados . . . . .	97
A.1. Resultados de los experimentos con modelos extendidos. . . . .	101

# Capítulo 1

## Introducción

Hoy en día, la interacción entre el ser humano y las computadoras se encuentra presente en todos los ámbitos de la vida cotidiana, ya sea para consultar el correo electrónico, hacer el pago de servicios o simplemente jugar. Una de las formas en la que este escenario se presenta es a través de los sistemas de diálogo. En la actualidad esta forma de interacción no es tan común, pero se espera que en un futuro sea parte de nuestra vida diaria. El desarrollo de sistemas de diálogo se ha visto impulsado por el aumento de la capacidad de cómputo. Los sistemas de diálogo son programas de cómputo que tienen la finalidad de mantener una conversación coherente y estructurada con una persona. Sin embargo, las interacciones entre estos sistemas y los humanos se ven limitadas por diferentes aspectos y situaciones. Uno de estos aspectos es el deficiente desempeño durante el entendimiento de las interacciones que los sistemas de diálogo reciben del exterior, ya que sin una correcta interpretación de las mismas, éste no será capaz de dar una respuesta adecuada.

El trabajo planteado en esta tesis se encuentra enmarcado dentro del proyecto Golem-UNIVERSUM, durante el cual se ha desarrollado un sistema de diálogo

que tiene como función principal el interactuar con personas a través de un juego de preguntas y respuestas. Este sistema presenta algunos problemas durante las interacciones, debido a las limitantes que presenta al realizar el proceso del entendimiento del lenguaje natural, ya que no es capaz de interpretar de manera correcta algunas de las expresiones emitidas por el usuario, provocando que la capacidad de respuesta del sistema se vea limitada. La presente tesis se plantea con el propósito de mejorar el entendimiento del lenguaje natural en el sistema Golem-UNIVERSUM. Para alcanzar el propósito planteado se propone el uso de una representación semántica<sup>1</sup> adecuada para el dominio en el que el sistema se desempeña, que sea capaz de maximizar el entendimiento del lenguaje en el módulo. Se plantea también el uso de un modelo gráfico no dirigido conocido como Lógica de Markov (Richardson and Domingos, 2006) para el modelado de las relaciones entre las representaciones semánticas y las elocuciones de entrada al sistema. El modelado con Lógica de Markov si bien se ha comenzado a utilizar con éxito relativo en el área de entendimiento de lenguaje hablado en el idioma inglés, no ha sido utilizado con este objetivo en el español. El desempeño del modelo obtenido será evaluado, con la finalidad de compararlo con el pársers semántico que actualmente se encuentra implementado en el sistema, para así determinar el modelo que mejor lleve a cabo la tarea del entendimiento del lenguaje natural.

El resto de este capítulo está conformado de la siguiente manera. En la sección 1.1 se presenta una breve introducción a los sistemas de diálogo y algunos ejemplos de los mismos, haciendo énfasis en sus módulos de entendimiento del lenguaje natural. En la sección 1.2 se explica de manera breve en que consiste el problema

---

<sup>1</sup>Una representación semántica es una estructura formal que expresa el significado de una expresión escrita o hablada.



del entendimiento del lenguaje natural. En la sección 1.3 se explica el problema del entendimiento del lenguaje natural en el sistema Golem-UNIVERSUM. En la sección 1.4 se enlistan los objetivos que se plantean para este trabajo. Finalmente, en la sección 1.5 se presenta la estructura general de esta tesis.

## 1.1. Sistemas de diálogo

Un sistema de diálogo es un programa de cómputo cuyo fin es mantener una conversación coherente y estructurada con una persona. Para poder mantener esta conversación, los sistemas de diálogo emplean diferentes modalidades de comunicación, tanto en el canal de entrada como en el canal de salida. Algunas de las modalidades más comunes que se emplean son el texto, la voz, los gestos y las imágenes. Los sistemas de diálogo se encuentran estructurados principalmente de tres etapas [ver Figura 1.1]. La primera de ellas es la *etapa de entendimiento de lenguaje*, la cual se encarga de decodificar o reconocer la entrada al sistema (e.g. en los sistemas conversacionales hablados se trata de la voz). La segunda etapa es el *manejador del diálogo*. Esta etapa es primordial en cualquier sistema de este tipo, su tarea es controlar el estado de la conversación, es decir, debe decidir cuál es la respuesta más apropiada que debe generar el sistema basándose en la entrada recibida. La última etapa, que conforma un sistema de diálogo, se trata de la *etapa de generación de la respuesta*, donde se exterioriza la decisión tomada por el manejador del diálogo (e.g. un sintetizador de voz).

En sus inicios, el desarrollo de los sistemas de diálogo era mucho más simple de lo que se puede observar en la actualidad. Los primeros sistemas que se desarrollaron eran conocidos como compañeros de conversación. Un ejemplo de este



Figura 1.1: Arquitectura de un sistema de diálogo.

tipo de sistemas es ELIZA (Jurafsky and Martin, 2008), desarrollado por el Instituto Tecnológico de Massachusetts (MIT, por sus siglas en inglés). El sistema ELIZA mantenía una conversación con el usuario por medio de cadenas de texto. El entendimiento del lenguaje natural en este sistema se llevaba a cabo buscando las palabras clave dentro de la cadena de entrada, con el fin de determinar la respuesta más apropiada; en caso de que no contara con una, el sistema respondía al usuario con la misma expresión pero a manera de reflexión o comentario.

Otro ejemplo de un sistema compañero de conversación es PARRY (McTear, 2004), desarrollado por la Universidad de Stanford. El sistema PARRY implementaba un modelo del comportamiento de una persona paranoico esquizofrénica, lo cual causó que este sistema presentara un grado más complejo en su desarrollo comparado con ELIZA. El sistema PARRY se comunicó con el sistema ELIZA por medio de la red ARPANET. Durante esta interacción, el sistema ELIZA interpretó el papel del psicoterapeuta del sistema PARRY, ya que tenía implementado un modelo para este fin. El entendimiento del lenguaje natural, en este sistema se llevaba a cabo buscando las palabras clave dentro de la cadena de entrada.

Poco tiempo después, la complejidad de los sistemas se vio incrementada con el aumento de la capacidad de cómputo de los componentes usados para su desarrollo, como es el caso del Sistema de Información del Lenguaje Natural en Ciencias Lunares, mejor conocido como LUNAR (Woods, 1973). El sistema LUNAR tenía

como objetivo el responder a consultas sobre los análisis geológicos realizados a las rocas que se obtuvieron durante las misiones Apolo a la luna. Durante el desarrollo del sistema se pretendía que la máquina se adaptará a las convenciones del idioma inglés. Esto último con la finalidad de agilizar las consultas realizadas a la base de datos de los análisis por investigadores, cuya lengua nativa pudiera o no ser el inglés. Con el fin de entender las consultas realizadas, el sistema contaba con un conjunto de reglas de interpretación semántica, además de un diccionario con 3500 palabras aproximadamente. Una característica importante del sistema LUNAR es que fue de los primeros en realizar un análisis léxico antes de llevar a cabo el entendimiento del lenguaje natural.

Con el paso del tiempo, otras características fueron apareciendo en los sistemas de diálogo. Por ejemplo, aparecieron diferentes modalidades de entrada, como el lenguaje hablado, además los sistemas empezaron a contar con base de conocimientos útiles para las interacciones. También, se desarrollaron tecnologías de síntesis de voz, y con esto se da paso a los sistemas conversacionales como se conocen en la actualidad.

Los sistemas de diálogo conversacionales actualmente son utilizados para resolver problemas en distintos aspectos de la vida cotidiana, por lo que su uso se encuentra más generalizado de lo que se puede pensar. Un ejemplo de este tipo de sistemas son los GPS instalados en los automóviles, los cuales reciben instrucciones por medio de la voz y responden mediante síntesis de la misma.

Los sistemas de diálogo conversacionales han encontrado en el sector de los servicios un área en la que sus características han sido aprovechadas de manera exitosa. Un ejemplo de ello es el sistema GALAXY (Seneff et al., 2003), desarrollado por el MIT en 1994. Este sistema cuenta con una interfaz de lenguaje hablado en

idioma inglés para obtener información en línea. Inicialmente fue desarrollado en un dominio relacionado a viajes y podía proporcionar información referente a viajes de avión, ubicación en algunas ciudades y el clima. El sistema GALAXY emplea un marco de representación semántica para el entendimiento del lenguaje natural. La información del marco se divide en tres campos: oración, tema y predicado.

Otros ejemplos de sistemas de diálogo utilizados en el sector de los servicios son el sistema TRAINS (Allen et al., 2000) y el sistema TRIPS (Ferguson and Allen, 1998), ambos proyectos desarrollados por la Universidad de Rochester. El sistema TRAINS está diseñado para la planeación de viajes en tren, para lo cual el usuario cuenta con una pantalla que despliega un mapa con las líneas férreas existentes. El usuario principalmente plantea preguntas de logística que el sistema responderá. Para llevar a cabo esta tarea, el sistema emplea el entendimiento del lenguaje natural con un procesamiento basado en reglas.

El sistema TRIPS es el último de una serie de prototipos de asistentes de planificación desarrollados por Departamento de Ciencias de la Computación. El objetivo del proyecto es el desarrollo de un asistente de planificación inteligente que interactúa con personas usando una combinación de lenguaje natural y gráficos (mapas, gráficos, etc.). Para el desarrollo del sistema TRIPS se define a la interacción como un diálogo entre el sistema y el humano. El diálogo proporciona el contexto para interpretar las expresiones y acciones humanas, y proporciona la estructura para decidir qué hacer en respuesta. Este sistema trabaja sobre diferentes dominios, entre los cuales se encuentran Pacifica<sup>2</sup>, AirLift<sup>3</sup> y el TRIPS-911<sup>4</sup>. Este sistema implementa un módulo de entendimiento del lenguaje natural basado

---

<sup>2</sup><http://www.cs.rochester.edu/research/trips/domains/pacifica.html>

<sup>3</sup><http://www.cs.rochester.edu/research/trips/domains/airlift.html>

<sup>4</sup><http://www.cs.rochester.edu/research/trips/domains/trips911.html>

en reglas.

Los sistemas de diálogo no sólo son utilizados en el sector de los servicios, sino también han encontrado cabida en el área de la investigación. Un ejemplo de un sistema de diálogo utilizado con este fin es el sistema August (Gustafson et al., 1999). Este sistema está desarrollado por el CTT–Centre for Speech Technology en Suecia. El objetivo de este sistema es estudiar el comportamiento de los usuarios con un escaso o nulo conocimiento acerca de sistemas de diálogo. El sistema utiliza un agente animado que representa la cabeza del escritor August Strindberg, el cual se comunica con los usuarios por medio de voz, expresiones faciales y movimientos de cabeza mientras que el usuario se comunica con el agente por medio de voz únicamente. Una de las características más importantes de este sistema es que trabaja en diferentes dominios, ya que se le puede preguntar acerca de servicios prestados en Estocolmo, de las investigaciones realizadas en el CTT o del sistema mismo. Esta última característica es de mucha importancia, ya que se desea captar la mayor cantidad de frases espontáneas dichas por el usuario, con la finalidad de estudiar su comportamiento. El módulo mediante el cual se lleva a cabo el entendimiento del lenguaje natural está basado totalmente en ejemplos, es decir, no se emplearon reglas ni procesamientos semántico o gramático.

Otro ejemplo de un sistema de diálogo desarrollado en el área de la investigación es el sistema Golem-UNIVERSUM como parte del proyecto con el mismo nombre. El proyecto es realizado por el departamento de Ciencias de la Computación (DCC) del Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas (IIMAS) y el Museo de las Ciencias UNIVERSUM, ambos pertenecientes a la Universidad Nacional Autónoma de México (UNAM). El proyecto Golem-UNIVERSUM se encuentra contextualizado en el área de inteligencia artificial; consiste en un

módulo que implementa un sistema conversacional que interactúa con los usuarios por medio de reconocimiento y síntesis de voz, comprensión del lenguaje hablado, así como el reconocimiento de imágenes (Meza et al., 2010).

El proyecto Golem-UNIVERSUM tiene como principal objetivo el acercar al público en general al uso de sistemas de diálogo conversacionales. El sistema conversacional implementado es un juego que está diseñado para que el usuario interactúe con el sistema mediante una serie de preguntas y respuestas. El propósito del juego es que el usuario adivine una carta, seleccionada previamente por el sistema de un conjunto de 10 cartas, a través de una serie de preguntas sobre las características presentes en las imágenes impresas en las cartas empleadas para jugar. Durante toda la interacción entre el usuario y el sistema la principal modalidad de comunicación es la conversación mediante la voz, lo que implica el reconocimiento y la síntesis de la misma, además del despliegue de imágenes. Dentro del módulo hay una mesa sobre la cual se encuentran 10 cartas que tienen impresos objetos relacionados con la astronomía. La aplicación está dirigida principalmente a niños de edades entre los 10 y los 16 años, sin que esto sea limitante para que personas de mayor edad puedan jugar.

Los ejemplos de sistemas de diálogo descritos muestran que el entendimiento del lenguaje natural dentro de mencionados sistemas ha sido desarrollado principalmente de dos formas. La primera de ellas es el análisis para obtener las palabras clave que están presentes en la entrada del sistema (voz o texto), mientras que la segunda se basa en la comparación de la entrada con patrones semánticos o reglas, mediante las cuales se puede determinar el significado de la entrada. El entendimiento del lenguaje natural es uno de los principales retos que se tiene en el desarrollo de sistemas conversacionales y es de suma importancia para asegurar

que un sistema pueda alcanzar el objetivo para el que fue desarrollado.

## 1.2. Procesamiento semántico

El nivel del entendimiento en un sistema de diálogo depende principalmente de la comprensión de las elocuciones de entrada. Una elocución es una unidad de discurso completa en lengua hablada, es decir, una elocución en el lenguaje hablado es lo que una oración en el lenguaje escrito. Para poder entender las elocuciones de entrada el sistema conversacional las somete a un procesamiento semántico, el cual consiste en encontrar el significado de la entrada al sistema. Este proceso identifica y asigna una representación semántica a la elocución, basándose en la idea de que el significado de la misma puede ser capturado en estructuras formales. La necesidad de utilizar estas representaciones aparece porque la transcripción de la elocución de entrada no es suficiente para realizar un procesamiento semántico correcto. Además de que muchas de las tareas que un sistema de diálogo puede ejecutar son determinadas por el significado de la entrada.

Como se mencionó con anterioridad, la comprensión del lenguaje hablado que el sistema conversacional del módulo Golem-UNIVERSUM, utiliza como entrada, es uno de los principales retos que actualmente se presenta en su desarrollo. El problema de la comprensión del lenguaje consiste en dar un significado a la expresión identificada por el reconocedor de voz. La forma en la que el sistema identifica las expresiones de entrada es transformando la señal acústica de la expresión a texto. El reconocedor de voz es el encargado de realizar esta transcripción, con la cual se inicia el proceso de comprensión. La comprensión se lleva a cabo mediante la asignación de significado a la transcripción, utilizando para esto una representación

semántica. La representación es enviada al manejador del diálogo, para que éste pueda determinar cuál es la respuesta adecuada a la entrada del sistema. Actualmente el sistema Golem-UNIVERSUM tiene una limitada capacidad de respuesta, debido a que presenta errores durante la asignación de representaciones semánticas a las elocuciones de entrada. El procesamiento semántico o la tarea de asignar una representación semántica a la elocución de entrada del sistema conversacional en el módulo Golem-UNIVERSUM es realizada por un pársers semántico. Este pársers forma parte de la etapa de entendimiento del lenguaje del sistema [ver Figura 1.2]. Dentro de esta etapa del sistema, el módulo del reconocedor de voz se encarga de transcribir la elocución de entrada a una cadena de texto, a su vez, el módulo del pársers semántico se encarga de asignar una representación semántica a la cadena.

El procesamiento semántico es una tarea complicada en su ejecución debido a que el sistema debe asignar una representación que exprese de manera adecuada la intención de la elocución. En caso de que el significado no sea entendido de manera correcta, se provoca un desempeño erróneo en el sistema, ya que la respuesta dada no será la adecuada.

El entendimiento del lenguaje hablado puede ser resuelto por medio de diferentes enfoques, como se verá a detalle en el capítulo 2. Uno de estos es el enfoque basado en análisis, en el cual el desarrollador del sistema debe diseñar e implementar un módulo del pársers semántico. El módulo debe contener patrones léxicos o semánticos que permiten construir una representación semántica. Esto implica que durante el desarrollo del módulo se deben contemplar todas las posibles elocuciones de entrada que se puedan presentar, lo cual a su vez es complicado para un sistema conversacional porque es muy difícil predecir cuáles serán las expresiones dadas por el usuario. Los sistemas LUNAR, TRIPS y otros de los mencionados



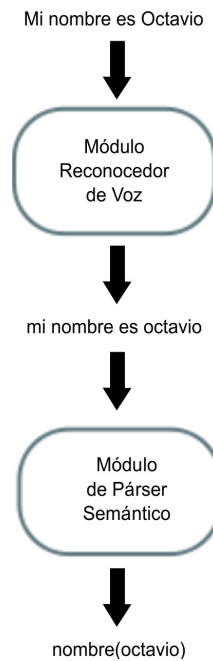


Figura 1.2: Módulos de la etapa de entendimiento del lenguaje en el módulo Golem-UNIVERSUM.

anteriormente realizan un procesamiento semántico basado en análisis.

Otro de los enfoques por medio del que se puede asignar una representación semántica a la elocución de entrada está basado en ejemplos. En este enfoque se requiere el uso de un corpus<sup>5</sup>, el cual sirve para modelar las relaciones entre las elocuciones de entrada y las representaciones semánticas. El modelado de las representaciones es considerado un proceso estocástico.

Ambos enfoques presentan ventajas y desventajas, como son la portabilidad o la capacidad de trabajar con palabras que no han sido utilizadas en el modelado. Lo anterior debe ser tomado en cuenta, ya que su aplicación se debe determinar considerando las características y necesidades del sistema conversacional al que se

---

<sup>5</sup>Un corpus es un conjunto, normalmente muy amplio, de ejemplos reales de un fenómeno.

desea integrar.

Para efectos de este trabajo se propone el uso de una técnica basada en ejemplos para el modelado de las relaciones entre las elocuciones de entrada al sistema y la representación semántica que mejor describe su intención. Para el entrenamiento del modelo se emplea el corpus Golem-UNIVERSUM, que ha sido recolectado durante las demostraciones diarias del sistema en el Museo de las Ciencias UNIVERSUM.

### **1.3. Objetivos**

El objetivo del presente trabajo es mejorar el desempeño del sistema Golem-UNIVERSUM, mediante la implementación de un pársers semántico que ayude a mejorar el entendimiento de las elocuciones de entrada del sistema. Se propone el uso de Lógica de Markov para modelar las relaciones entre las elocuciones de entrada y las representaciones semánticas.

#### **1.3.1. Objetivos específicos**

- Proponer un modelo en Lógica de Markov robusto que relacione las transcripciones de las elocuciones con las representaciones semánticas adecuadas.
- Entrenar los modelos propuestos con la información que proporciona el corpus Golem-UNIVERUSM.
- Seleccionar las métricas de evaluación que permitan un análisis correcto de los resultados.
- Evaluar el desempeño de los modelos entrenados.

- Implementar el modelo en el sistema Golem-UNIVERSUM.

## 1.4. Estructura de la tesis

El capítulo 2 presenta los trabajos previos realizados en el área de procesamiento semántico, así como una descripción del proyecto Golem-UNIVERSUM. El capítulo 3 describe, de manera detallada, la solución propuesta para mejorar el desempeño del pársers semántico en el sistema Golem-UNIVERSUM, así como los pasos necesarios para llevarla a la práctica. El capítulo 4 muestra los experimentos realizados utilizando el corpus Golem-UNIVERSUM, con la finalidad de obtener el modelo que mejor desempeñe el entendimiento del lenguaje natural. Se presentan los resultados obtenidos de los experimentos. El capítulo 5 describe los experimentos utilizando el modelo obtenido en el capítulo anterior, incluyendo en éstos el contexto de la oración, así como los resultados correspondientes. El capítulo 6 presenta las conclusiones de este trabajo, las contribuciones hechas por el mismo, así como futuras líneas de trabajo.



## Capítulo 2

# Trabajo previo

El entendimiento del lenguaje hablado, en años recientes, ha sido un factor muy importante para el desarrollo de sistemas de diálogo conversacionales. Esto se debe principalmente a que su implementación ha dado como resultado sistemas con la capacidad de sostener conversaciones más fluidas y coherentes. Con el objetivo de mejorar y optimizar el proceso del entendimiento del lenguaje en los sistemas conversacionales, se han realizado investigaciones para aplicar diferentes técnicas y procedimientos en este proceso. Los resultados obtenidos de estas investigaciones han sido muy variados. Algunas técnicas han mostrado un buen desempeño de la tarea, aún al momento de realizar el procesamiento de datos inesperados, mientras que otras técnicas no tienen la capacidad de trabajar con datos desconocidos.

El contenido de este capítulo tiene como objetivo mostrar los antecedentes que se tienen para el desarrollo de este trabajo. Para alcanzar dicho objetivo este capítulo está dividido en dos secciones. La sección 2.1 presenta las diferentes formas en las que se puede realizar el procesamiento semántico. En la sección 2.2 se presenta el proyecto sobre el cual se desarrolla esta tesis. Es en esta misma sección donde

se explica de manera precisa el problema que se plantea, de qué manera afecta al desempeño del sistema en el módulo, además de la forma en la que la versión actual del sistema lleva a cabo el procesamiento del lenguaje hablado.

## 2.1. Estado del arte en procesamiento semántico

Como se ha mencionado en el capítulo 1, existen diferentes técnicas por medio de las cuales se puede dar solución al problema del entendimiento del lenguaje hablado. Estas técnicas se encuentran categorizadas en dos grandes enfoques: el basado en ejemplos y el basado en análisis. En esta sección se describen principalmente los métodos del enfoque basado en ejemplos, específicamente los modelos gráficos, categoría en la que se encuentra el método que se plantea como solución y que es descrito en el capítulo 3.

Los métodos basados en ejemplos tienen la característica principal de utilizar un corpus para poder modelar el procesamiento semántico como un proceso estocástico. Un modelo gráfico es un modelo probabilístico asociado a un grafo, el cual denota la dependencia condicional entre variables aleatorias. Estas relaciones son de suma importancia para estos modelos, ya que relacionan variables observadas, que son los datos que se conocen y a los que se tiene acceso, con variables ocultas, que son aquellos datos que se desean obtener o saber. Un ejemplo sencillo de esto se presenta en la Figura 2.1, donde se desea etiquetar a las palabras de una oración con la categoría gramatical a la que pertenecen. En este ejemplo las variables observadas son las palabras de la oración que se analiza (indicadas con un círculo color gris). Las variables ocultas son las etiquetas de las categorías gramaticales que se espera sean asignadas a las palabras (indicadas con un círculo

color blanco). Este ejemplo será utilizado a lo largo del capítulo para mostrar la manera en la que se desempeñan los diferentes métodos que se describen.

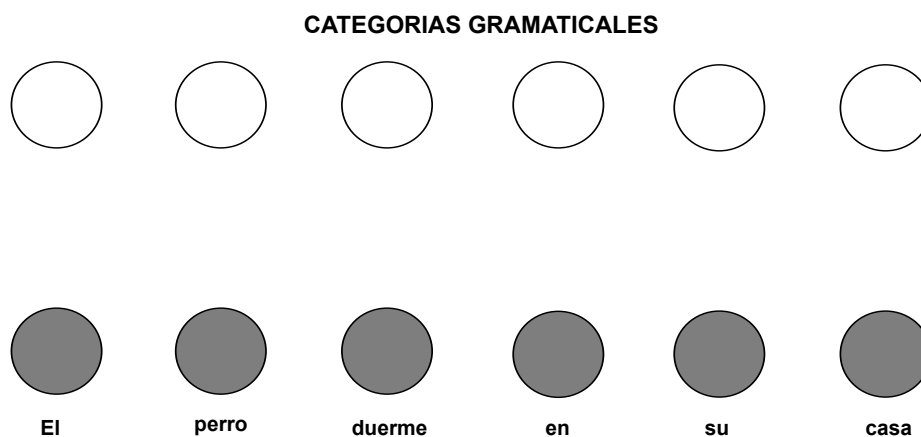


Figura 2.1: Ejemplo de variables ocultas y variables observadas.

Los modelos gráficos aparecieron en la década de los 80's principalmente en el área de aprendizaje automático. Estos modelos presentan algunas características que los separan de otros modelos, como el análisis computacional relacionado con la teoría de las probabilidades, descripción de las distribuciones de probabilidad de las observaciones y, una muy importante, el poder expandir el análisis empleando nuevos elementos u observaciones. Las distribuciones de probabilidades de estos modelos son utilizadas en el procesamiento del lenguaje natural para modelar diferentes tareas, como la distribución de palabras en un párrafo, la estructura gramatical de estas palabras, la traducción automática o tareas de etiquetado. Como se comentaba el grafo sirve para determinar la dependencia entre las variables, además la estructura del mismo nos permite la consulta de las probabilidades que se presentan por medio de la inferencia. Algunos de los valores de probabilidad que se pueden calcular empleando la inferencia incluyen:

- La probabilidad de que cierta variable tome un valor específico.
- Encontrar cuál de los valores que puede tomar una variable es el más probable.
- La probabilidad de una secuencia de variables ocultas.
- Dada una secuencia de variables observadas, cuál es la secuencia de variables ocultas con la que tiene mayor probabilidad de estar relacionada.

Cabe mencionar que los cálculos no sólo pueden reportar probabilidades, sino en caso de que se trabaje con un corpus indicar cuáles son los parámetros para un modelo gráfico en específico.

Existe dentro de los modelos gráficos varias clasificaciones conforme a la forma en la que el grafo está definido. Una de las categorías es la de los modelos dirigidos, también conocidos como redes bayesianas (Cohn, 2007). Los grafos en estos modelos están definidos como  $G = (X, E)$ , donde  $X = \{X_1, X_2, X_3 \dots X_N\}$  son los nodos, y  $E = \{(X_i, X_j)\}$  son las aristas. Los nodos para esta definición se refieren a las variables, y las aristas representan las posibles relaciones entre dichas variables. Por la definición de los grafos utilizados en estos modelos, las probabilidades empleadas son probabilidades condicionales. Los modelos ocultos de Markov o HMM's (por sus siglas en inglés, Hidden Markov Models) (Rabiner, 1989) son un ejemplo de modelos en esta categoría. Los HMM's son modelos estadísticos en los que se asume que el sistema a modelar es un proceso de Markov de parámetros desconocidos, su objetivo es determinar los valores de las variables ocultas a partir de las observadas. Los valores extraídos se pueden emplear para llevar a cabo análisis sucesivos, por ejemplo en aplicaciones de reconocimiento de patrones. Un HMM se



puede considerar como la red bayesiana dinámica más simple. El ejemplo de etiquetado mediante HMM puede ser mostrado gráficamente, tal y como se muestra en la Figura 2.2. Por la forma en la que el grafo está conformado, específicamente por las flechas que representan las relaciones entre las variables, se puede observar que se trata de un cálculo de probabilidades condicionales. El cálculo de la distribución de probabilidad de la variable oculta depende únicamente de la probabilidad de la variable oculta anterior. Y para el caso de la variable observada, el valor de probabilidad de ésta sólo depende del valor de la variable oculta.

El modelo matemático de la probabilidad de una secuencia de variables observables  $X = x(0), x(1), \dots, x(L - 1)$  de longitud  $L$ , esta dada por la fórmula 2.1.

$$P(X) = \sum_Y P(X|Y)P(Y) \quad (2.1)$$

donde la sumatoria se extiende sobre todas las posibles secuencias de las variables ocultas  $Y = y(0), y(1), \dots, y(L - 1)$ . Para el caso de las secuencias de variables mostradas en la Figura 2.2, la secuencia tiene una longitud  $L = 6$ .

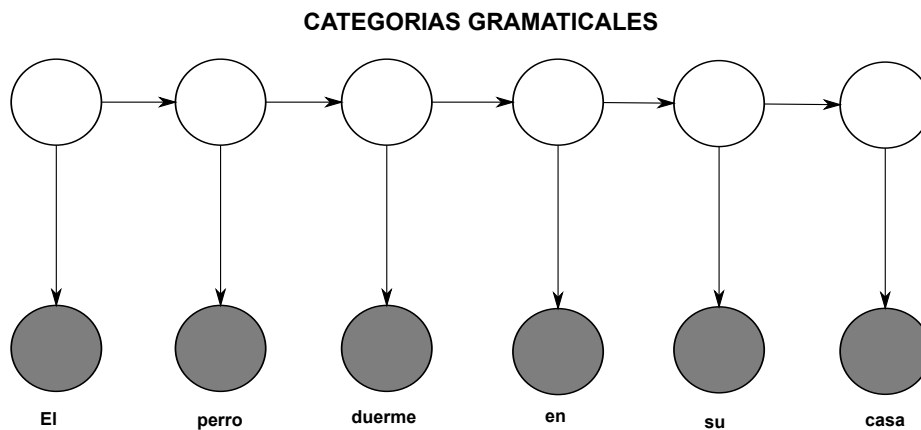


Figura 2.2: Ejemplo de etiquetado utilizando HMM's.

La otra categoría existente dentro de los modelos gráficos es la de los modelos no dirigidos (Cohn, 2007). El grafo no dirigido está definido como  $G = (X, E)$ , donde  $X = \{X_1, X_2, X_3 \dots X_N\}$  son los nodos, y  $E = \{(X_i, X_j) : i \neq j\}$  son las aristas. Con esta definición se puede decir que las probabilidades empleadas en estos modelos son probabilidades conjuntas. Una técnica de los modelos gráficos no dirigidos utilizados para resolver el problema del procesamiento semántico es conocida como de máxima entropía o MaxEnt. MaxEnt es un marco para la integración de información de diferentes fuentes con la finalidad de llevar a cabo una clasificación. El conjunto de características, por medio de las cuales se describe el conjunto de datos utilizados para esta clasificación, permite usar conocimiento *a priori* para delimitar que tipo de información es importante para la clasificación. Dentro del modelado con MaxEnt, la selección de características y el entrenamiento del modelo se encuentran integrados, lo que permite especificar desde el principio la información que será relevante para el modelado. Para una característica, primero se obtiene la probabilidad de cada uno de los valores asociados a la misma. Este cálculo se realiza con base en el conjunto de entrenamiento. Obtenidas todas las probabilidades, se procede a determinar cuál de todos los valores presenta la mayor probabilidad, lo que indica que tiene la mayor entropía y, por lo tanto, el valor que se debemos tomar para esa característica (Manning and Schütze, 1999). Haciendo referencia al ejemplo de etiquetado mostrado en la Figura 2.1, el modelado con MaxEnt se presenta de la manera indicada en la Figura 2.3.

La expresión del modelo matemático de MaxEnt se presenta en la ecuación 2.2.

$$p(y|x) = \frac{1}{Z} \exp \left( \sum_{i=1}^n \lambda_i f_i(y, x) \right) \quad (2.2)$$

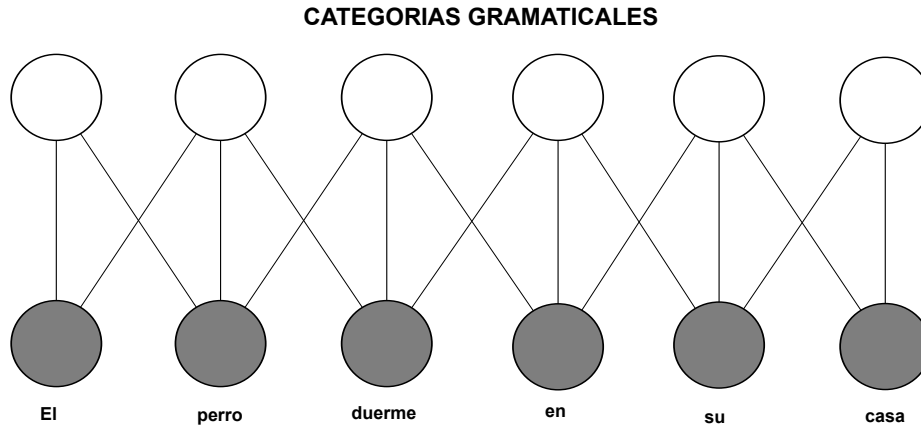


Figura 2.3: Grafo asociado a una tarea de clasificación.

donde

$$Z = \sum_y \exp \left( \sum_{i=1}^n \lambda_i f_i(y, x) \right) \quad (2.3)$$

Como se puede apreciar en la fórmula 2.2, la salida de este modelo es una variable y no un conjunto de ellas, ya que el modelo considera una variable a la vez, debido a que las considera independientes entre ellas. Una de las principales características de este modelo es la función de atributos, denotada por  $f_i$  en la ecuación. Estas funciones tienen como objetivo el denotar las relaciones entre la variable oculta y las observadas. Estas funciones retornan 1 en caso de que la relación este presente, o un 0 en caso contrario. Ese valor es multiplicado por el peso, denotado por  $\lambda_i$ . Con esto último, se puede decir que MaxEnt modela todo aquello que se considera conocido y que le es presentado mediante un corpus, pero así también no es capaz de asumir nada respecto a lo que es desconocido, es decir, no forma parte del corpus. Las relaciones con mayor probabilidad son aquellas que presentan un peso más grande comparado con otras (Ratnaparkhi, 1999).

Tomando en cuenta lo descrito, se pueden hacer visibles algunas diferencias entre HMM's y MaxEnt. La primera de ellas es que mientras el primero puede calcular probabilidades trabajando con secuencias de variables, ya sea observadas u ocultas, el modelo de MaxEnt sólo puede trabajar con una variable a la vez. Lo anterior nos muestra que con HMM's se pueden trabajar relaciones entre variables ocultas lo cual podemos percibir al momento de comparar los grafos asociados. Otra diferencia es la probabilidad que emplean, ya que mientras los modelos ocultos emplean una probabilidad condicional, MaxEnt emplea una probabilidad conjunta.

Otro ejemplo de los modelos gráficos no dirigidos son los campos condicionales aleatorios o CRF's (por sus siglas en inglés Conditional Random Fields). En estos modelos cada vértice del grafo asociado representa una variable aleatoria cuya distribución de probabilidad se deduce, y cada arista representa una dependencia entre dos variables aleatorias. Para el debate actual, se asume de entrada que la secuencia  $X$  representa una secuencia de observaciones y  $Y$  representa una variable oculta, cuyo valor debe ser deducido de las observaciones dadas. En un CRF, la distribución de cada variable aleatoria discreta  $Y$ , en el gráfico, está condicionada a una secuencia de entrada  $X$ . En la Figura 2.4 se muestra el ejemplo de etiquetado manejado durante esta sección resuelto con CRF's.

El modelo matemático de un CRF, para el etiquetado de una secuencia de  $X$ , se muestra en la ecuación 2.4

$$p_{\theta}(x|y) = \frac{\prod_{i=1}^{n+1} M_i(x_{i-1}, x_i|y)}{\left(\prod_{i=1}^{n+1}, M_i(y)\right)_{start, stop}} \quad (2.4)$$

donde  $x_0 = start$  y  $x_{n+1} = stop$ . Las funciones  $M_i$  están determinadas por las aristas que relacionan a las variables. En la representación gráfica del modelo los

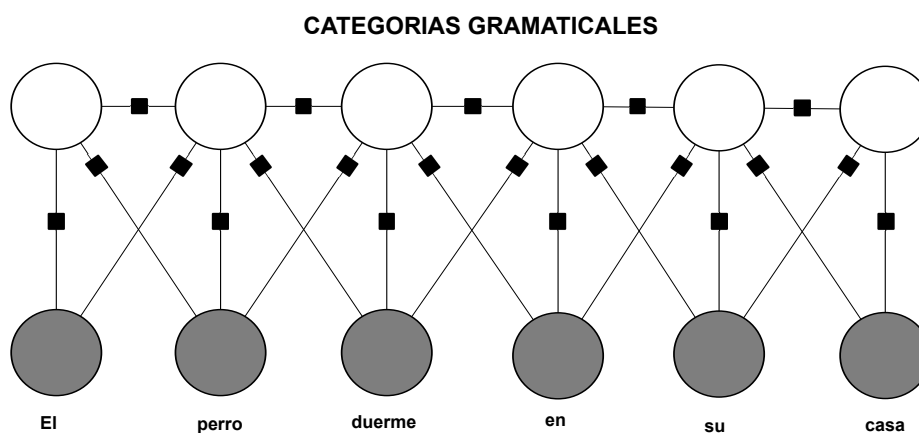


Figura 2.4: Grafo relacionado a un CRF para una tarea de etiquetado.

vértices representan a las variables (Lafferty et al., 2001).

Los campos condicionales aleatorios permiten descartar algunas relaciones de dependencia que se asumen en algunos modelos como los HMM's. Otra característica es que permiten las relaciones entre variables ocultas, además de las observadas, tal y como se puede ver en la Figura 2.4. Las relaciones mencionadas, nos dan la posibilidad de obtener modelos mejor adaptados.

Los métodos mencionados anteriormente han sido utilizados para realizar la tarea de procesamiento semántico en diferentes situaciones. Esto sirve también de antecedente para la Lógica de Markov, modelo que está incluido en la solución propuesta en el presente trabajo. El desarrollo de la propuesta mencionada se presenta de manera detallada en el siguiente capítulo. Lógica de Markov es un modelo que está incluido en la categoría de modelos gráficos no dirigidos, por lo que requiere de la existencia de un corpus para que sea posible el entrenamiento del modelo.

## 2.2. Proyecto Golem-UNIVERSUM

El módulo Golem-UNIVERSUM implementa un sistema conversacional, que tiene como objetivo mostrar la tecnología, desarrollada por el Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas (IIMAS) perteneciente a la Universidad Nacional Autónoma De México (UNAM), al público en general. Los desarrollos de este instituto se centran principalmente en infraestructura tecnológica para el desarrollo de sistemas de diálogo que utilicen el idioma español. Se pretende que el presente trabajo aporte información suficiente para aumentar la robustez de los sistemas desarrollados, específicamente en el área del entendimiento del lenguaje natural.

Este módulo se encuentra en exposición permanente dentro del Museo de las Ciencias UNIVERSUM, el cual pertenece también a la UNAM. El sistema implementado en el módulo tiene como función principal el interactuar con niños de entre 10 y 16 años por medio de un juego llamado “Adivina la carta”. El juego tiene como finalidad que el usuario determine cuál es la carta que el sistema escogió al principio de la interacción. Para poder determinar de qué carta se trata, el usuario realiza una serie de preguntas acerca de las características que presentan las imágenes impresas en las cartas diseñadas para esta aplicación. El módulo cuenta con una cabina que sirve para aislar al jugador y al sistema de los sonidos externos que puedan interferir en la conversación. Dentro de la cabina se encuentra una computadora con un micrófono y un mouse, como dispositivos de entrada, y con unas bocinas y un monitor como dispositivos de salida. Para que las personas que se encuentran fuera puedan saber lo que se dice durante la interacción, el módulo cuenta, en su parte externa, con una pantalla y bocinas. En el módulo también

se encuentran las diez cartas con las que es posible llevar a cabo el juego. Las cartas tienen impresos motivos astronómicos, porque el año en que el sistema fue desarrollado fue el año Internacional de la Astronomía (Meza et al., 2010).



Figura 2.5: Cartas utilizadas en el juego “Adivina la carta”.

### 2.2.1. Estructura del sistema

La arquitectura principal del sistema se centra en el concepto de especificación e interpretación de modelos de diálogo. Un modelo de diálogo es la representación de un protocolo presente dentro de una conversación. La manera en que este protocolo está definido es en base al análisis que se realiza de las conversaciones reales. Los modelos de diálogo son definidos en términos de un conjunto de situaciones y relaciones entre situaciones. Dentro del conjunto de situaciones existen una situación inicial y una situación final. Las relaciones están determinadas por dos elementos: las expectativas y las acciones. Las expectativas pueden ser intencionales o naturales. Cada expectativa tiene una acción correspondiente, la cuál se llevará a cabo

si la expectativa es cubierta (Pineda et al., 2010).

El sistema del módulo es un conjunto de agentes que trabajan en conjunto para poder llevar a cabo la interacción con el usuario [ver Figura 2.6]. El agente principal del sistema es el manejador del diálogo, el cual controlará el estado de la conversación basándose en los modelos de diálogo previamente descritos. El resto de los agentes es ejecutado después para poder tener la inicialización del sistema lista, y es en este momento que se inicia la interacción. Cuando el usuario produce algún sonido, después de que el sistema le hace una solicitud, éste es captado por el agente reconocedor. La función de éste es reconocer la secuencia de palabras asociadas a la elocución de entrada. La cadena con la secuencia de palabras es enviada al agente Interpreta voz, el cual tiene como objetivo determinar la intención del usuario, basándose en las palabras que han sido identificadas por el reconocedor. La intención del usuario se denota con una representación semántica que es enviada al manejador del diálogo, el cual basado en los modelos de diálogo determina cuál es la respuesta adecuada a la elocución de entrada. La respuesta es externada por medio de los agentes Sintetizador y Despliega Medios. El primero sirve para decirle al usuario frases por medio de la síntesis de voz, las cuales son enviadas por el manejador del diálogo como cadenas de caracteres. El agente Despliega Medios sirve para mostrar contenido multimedia en la pantalla. Para poder determinar si la carta mostrada por el usuario es la correcta se utilizan el agente Reconocedor Visual y el agente Interprete Visual. El primero está encargado de tomar la fotografía de la carta en el momento en el que el usuario la muestra. La función del Interprete Visual es el determinar de qué carta se trata, para que basado en esta información el agente Manejador del Diálogo pueda responder al usuario. Todos los agentes utilizan al agente Facilitador del Diálogo



para poder comunicarse entre ellos.



Figura 2.6: Arquitectura del sistema implementado en el módulo Golem-UNIVERSUM.

Para la implementación de los agentes se utilizaron diferentes lenguajes de programación y herramientas. Para la comunicación de los agentes se utiliza el ambiente Open Agent Architecture (OAA) (Cheyer and Martin, 2001), ya que presenta muchas ventajas para relacionar el reconocimiento, la interpretación y la inferencia. El lenguaje de programación Prolog es utilizado para implementar el agente manejador del diálogo. El módulo Reconocedor implementa un reconocedor automático de voz basado en el software Sphinx3 (Huerta et al., 1999). Para el desarrollo de dicho reconocedor se llevó a cabo la recolección del corpus DIMEx100 en su versión para niños, debido al dominio del sistema desarrollado. Este corpus está basado en el corpus DIMEx100 adultos (Pineda et al., 2009). El agente Interpreta Voz emplea un análisis de palabras claves para poder determinar la intención

del usuario así como expresiones regulares para comunicarle al manejador la intención. El algoritmo SURF (Speeded-Up Robust Features) (Bay et al., 2008) basado en OpenCV (Bradski and Kaehler, 2008) es utilizado para el agente reconocedor visual y el intérprete visual (Pineda et al., 2010).

### **2.2.2. Procesamiento semántico**

El entendimiento del lenguaje natural en el sistema Golem-UNIVERSUM se lleva a cabo mediante la asignación de representaciones semánticas a las elocuciones de entrada. El pársers semántico con el que trabaja la versión actual del sistema recibe como entrada la transcripción de la elocución obtenida por medio del reconocedor de voz, así como la expectativa que el sistema tiene. La expectativa se refiere a que si el sistema solicita algún dato en específico, espera que el usuario responda a esta solicitud, en caso de que la respuesta no sea adecuada, el sistema la descarta y solicita nuevamente el dato. El pársers semántico cuenta con un conjunto de patrones, correspondientes a cada una de las expectativas que se tiene, y a su vez, cada patrón cuenta con un conjunto de palabras claves que pueden aparecer para cada expectativa. En caso de que alguna de las palabras de la transcripción de la elocución se encuentre dentro del conjunto de palabras clave del patrón, se asigna la representación semántica representada por ese patrón, complementándola con la palabra clave, si el patrón así lo requiere. El caso de las expectativas de una respuesta positiva o negativa es un ejemplo donde la expresión regular no se ve complementada por la palabra clave. Por el contrario, las expectativas de nombre, característica y edad, si requieren ser complementados con la palabra clave encontrada en la transcripción. En el Cuadro 2.1 se muestran las diferentes expectativas que se tiene, así como las expresiones regulares y las palabras claves relacionadas

Expectativa.	Expresión regular.	Palabras clave
característica	característica(X)	rojo, naranja, redondo, onda, tubo, planeta, estrella, gira
edad	edad(X)	diez, once, doce, trece, catorce, quince, dieciséis
afirmación	ok	ok, si, claro, bien, bueno, favor, afirmativo
negación	no	no, negativo, nel, tampoco
nombre	nombre(X)	brenda, carlos, juan, pedro, antonio, mariana, carla, camila, kevin

Cuadro 2.1: Ejemplo de expectativas, patrones y palabras claves en el sistema Golem-UNIVERSUM.

a cada una de ellas.

Durante el procesamiento semántico en el sistema Golem-UNIVERSUM se pueden presentar diferentes situaciones. Para una mejor comprensión de lo mencionado se presenta el Cuadro 2.2 como un auxiliar. En el cuadro, el título de la columna *Tur.* indica el número de turno que se presenta en ese momento de la conversación. La abreviatura *Hab.* es empleada para indicar cuál de los sujetos involucrados en la interacción le toca el turno de hablar: la letra *S* hace referencia al sistema y la letra *U* al usuario. En la columna *Inter.* se presenta la interpretación dada por el sistema a la elocución de entrada.

La situación de que los datos proporcionados por el usuario no sean adecuados, descrita anteriormente, se muestra en el turno 17 del Cuadro 2.2. En este turno el sistema está esperando encontrar una palabra referente a las características de las cartas. En este caso se muestra que el reconocedor detectó una palabra que fue “aldo”, pero como la expectativa que se tiene es de una característica, el sistema le

indica al usuario que no le entendió. Otra de las situaciones que se puede presentar durante las interacciones se muestra en el turno 3 de la iteración del cuadro 2.2, donde se indica que el sistema está solicitando el nombre del usuario. En este caso, la respuesta que el usuario da al sistema, se interpretará como un nombre y será enviada al manejador del diálogo, como la expresión regular mostrada en el turno 4 del mismo cuadro. Otra de las situaciones que es posible se presenten durante la iteración es la mostrada en el turno 23 del cuadro 2.2, donde el usuario pregunta si el color azul está presente en la carta escogida por el sistema. El sistema no capta de manera correcta esta intención, y por consiguiente el manejador del diálogo determinará la respuesta en base en lo que recibió. La respuesta es la mostrada en el turno 24.

El procesamiento semántico en el sistema del juego “Adivina la carta” se desarrolla dentro de la etapa de interpretación-especificación, en la sección de interpretación de la arquitectura mostrada en la Figura 2.7. Como se muestra en la Figura, la arquitectura del sistema está dividida en tres capas de acuerdo a (Pineda et al., 2010). La etapa de reconocimiento-representación, la de interpretación-especificación y por último la etapa de representación-inferencia.

### 2.2.3. Representaciones semánticas

En una sesión, es el sistema quien inicia la interacción presentándose y solicitando al usuario su nombre con la pregunta *¿Cuál es tu nombre?* como se muestra en el turno 1 del Cuadro 2.2. El usuario, en la mayoría de los casos, responde de una manera similar a la indicada en el turno 4 del mismo cuadro. Basándose en

Tur.	Hab.	Interacción/Hipótesis de reconocimiento/	Inter.
1	S	Hola a todos me llamo Golem y éste es el juego de adivina la carta, tenemos jugador, ¿Cuál es tu nombre?	
2	U	Octavio/ <i>one</i>	<i>notU</i>
3	S	Estoy medio sordo,¿Cuál es tu nombre?	
4	U	Octavio/ <i>hector luz</i>	<i>nombre(hector)</i>
5	S	Hola Hector, me da gusto jugar contigo/	
6	S	¿Me puedes decir tu edad?	
...			
17	U	¿Es roja?/ <i>aldo</i>	<i>notU</i>
18	S	Me puedes preguntar otra característica que veas por favor	
...			
24	S	No atinaste, en mi carta nada tiene luz propia. Okay, ésta es la última pregunta que me puedes hacer, piénsala bien.	
25	U	¿Es rojo?/ <i>es rojo</i>	<i>caracteristica(rojo)</i>
26	S	Ve algo parecido, pero no es rojo	
27	S	Por favor muéstrame la carta	
28	U	<b>El usuario muestra la carta del sol a la cámara</b>	<i>carta(sol)</i>
29	S	No tengo muy buena vista. ¿Es ésta la carta que me mostraste?	
30	U	Sí/ <i>si</i>	<i>ok</i>
31	S	Lástima, la carta que había escogido es la del planeta mercurio	
32	S	¡Oh! Pero antes de irte por favor mira a la cámara. Gracias por jugar conmigo, regresa pronto	

Cuadro 2.2: Extracto de un diálogo entre el usuario (U) y el sistema (S) (Tur., Turno; Hab., Hablante; Inter., Interpretación).

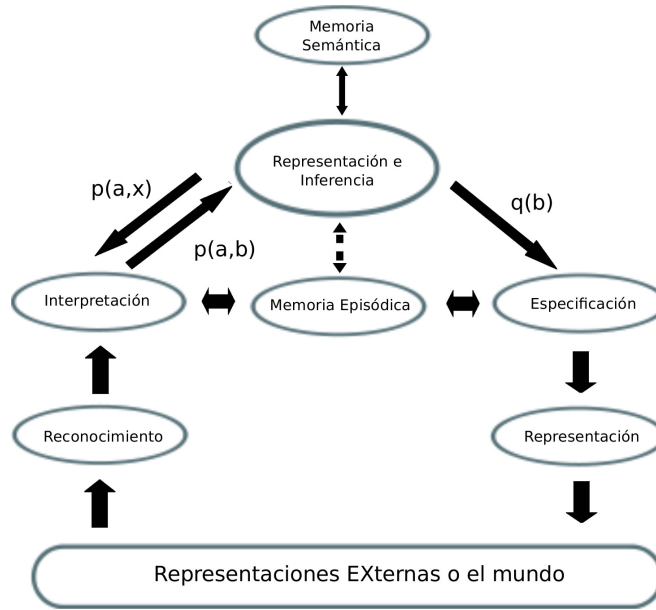


Figura 2.7: Arquitectura Cognitiva Orientada a la Interacción del sistema del juego “Adivina la carta”.

el análisis de esta elocución de entrada y dadas las expectativas que se tienen, el sistema emplea la representación semántica *nombre(octavio)* para representar el significado de esta elocución. La situación que se presenta en seguida es la solicitud de la edad del jugador mediante la pregunta *¿Me puedes decir tu edad?*, siendo una posible respuesta *trece años*. Para esta situación, la representación semántica empleada es de la forma *edad(trece)*. En caso de que la edad proporcionada por el usuario fuera menor al límite establecido, el sistema se disculpa e indica a la persona que no es posible continuar con el juego. En caso contrario, lo siguiente que el sistema realiza es preguntarle al usuario si desea que le sean explicadas las reglas del juego. Para las situaciones en que las respuestas dadas por el usuario sean afirmativas o negativas, se emplean las representaciones semánticas *ok* y *no*, respectivamente. En caso de que no sea requerida la explicación, el sistema da

inicio al juego, solicitándole al usuario su primer pregunta. Si las reglas son solicitadas al sistema, éste último explica de manera general la secuencia de pasos necesarios para jugar, además de algunos ejemplos de las preguntas que puede realizar. Después de terminada la explicación el sistema da inicio al juego. El formato de las preguntas es muy variado, por ejemplo *¿Se parece un plato?* y *¿Es azul?*, son algunas de las elocuciones de entrada que se presentan. Para esta situación, la representación semántica es de la forma *caracteristica(valor)*. Para las preguntas mostradas con anterioridad las representaciones semánticas utilizadas son *caracteristica(plato)* y *caracteristica(azul)*. Después de terminadas las cuatro preguntas que se permiten al usuario, el sistema le pide que muestre la carta que él supone es la elegida a una cámara. El sistema toma una fotografía de la carta mostrada, y realiza una comparación con la seleccionada. Para llevar a cabo la comparación, el sistema asigna una representación semántica a la carta reconocida, por ejemplo *carta(sol)* para la carta que presenta la imagen del Sol, o *carta(mercurio)* si se trata de la carta del planeta Mercurio. Basándose en el resultado de esta comparación el sistema responde al usuario sobre si la suposición hecha es correcta o no. Para finalizar la interacción se le pide al usuario que se coloque enfrente de la cámara del sistema para tomarle una fotografía. Cabe mencionar que el usuario no está obligado a realizar las cuatro preguntas, ya que puede indicarle al sistema de que carta se trata para poder dar paso a la toma de fotografía de la misma.

El contenido de este capítulo nos sirve como fundamento para el contenido del capítulo 3, donde se presenta la solución propuesta con el objetivo de mejorar el procesamiento semántico en el sistema Golem-UNIVERSUM. Además el capítulo 3 presenta las características del corpus a utilizar durante el desarrollo de este trabajo, así como las métricas de evaluación con las que el desempeño del modelo

será valorado.



## Capítulo 3

# Metodología propuesta

En el capítulo anterior, se mostraron diferentes técnicas empleadas para realizar la tarea del procesamiento semántico, las ventajas y desventajas que presentan, así como sus diferencias. El contenido de este capítulo presenta de manera precisa y detallada la solución propuesta para resolver el problema del procesamiento semántico en el sistema implementado en el módulo Golem - UNIVERSUM. Con la finalidad de proponer la mejor solución al problema del procesamiento semántico, se consideraron algunos aspectos. El primero de ellos es el desempeño del marco de Lógica de Markov cuando se emplea para llevar a cabo el procesamiento semántico (Meza, 2009). Este marco, si bien ha sido utilizado para modelar relaciones entre representaciones semánticas y elocuciones en idioma inglés, no ha sido utilizado con este mismo fin en el idioma español. Otro aspecto a considerar es la existencia de dos corpus con los cuales se llevará a cabo el entrenamiento de modelos. Ambos corpora son descritos a detalle más adelante. Otro aspecto a tomar en cuenta es que actualmente el sistema Golem-UNIVERSUM cuenta con un conjunto de representaciones semánticas que funcionan perfectamente para representar los significados

de las elocuciones de entrada, pero es la técnica de asignación la que presenta un bajo desempeño.

Este capítulo presenta la siguiente estructura. En la sección 3.1 se describen los corpora utilizados para el desarrollo de este trabajo, explicando sus características más importantes. En la sección 3.2 se presenta el concepto de Lógica de Markov así como una explicación del mismo. En la sección 3.3 se presentan las métricas que serán utilizadas para la evaluación de los modelos obtenidos, y así determinar cuál es el mejor modelo que lleva a cabo el procesamiento semántico.

### 3.1. Corpus Golem-UNIVERSUM

Al ser Lógica de Markov un modelo probabilístico basado en ejemplos requiere la existencia de un corpus para su entrenamiento. El corpus empleado en este trabajo es dependiente del dominio, ya que su recolección se llevo a cabo durante las interacciones entre los usuarios y el sistema del juego “Adivina la carta”, durante las demostraciones que se llevan a cabo en el Museo de las Ciencias UNIVERSUM.

El corpus consiste en transcripciones de las elocuciones de entrada al sistema, tanto automáticas como manuales, etiquetadas con una representación semántica. Las transcripciones manuales consisten en la secuencia exacta de palabras de la elocución que el usuario emitió durante la interacción transcritas por un ser humano. Por otro lado las transcripciones automáticas están formadas por la secuencia de palabras dada por el reconocedor de voz de la elocución emitida por el usuario. Algunos ejemplos de transcripciones, representaciones semánticas y expectativas de las elocuciones se presentan en el cuadro 3.1, donde la transcripción manual se muestra en *itálicas* y la transcripcion automática, la representación y la

---

<i>parece un ojo</i>
grande de un ojo
característica (ojo)
característica, carta
<i>tengo 14 años</i>
en en este tengo catorce
edad (14)
edad
<i>mi nombre es erik</i>
alambres erico
nombre (erik)
nombre
<i>si claro que si</i>
claro de lo es
ok
ok, no

---

Cuadro 3.1: Transcripciones, representaciones semánticas y expectativas de elocuciones del corpus Golem-UNIVERSUM.

expectativa se muestran en los renglones siguientes.

Para un mejor control de los experimentos que se realizan, el corpus se encuentra dividido en dos partes: desarrollo y evaluación. A continuación se proporcionan detalles técnicos de cada una de las partes.

### **Corpus de desarrollo**

El número total de elocuciones con las que cuenta el corpus en la parte de desarrollo es de 1806. Para el caso de las transcripciones manuales, el promedio de la longitud es de 1.73 palabras, mientras que la mediana es de 1 palabra. Las transcripciones manuales de las elocuciones utilizan un total de 3140 palabras. En lo que respecta a las transcripciones automáticas el promedio de la longitud es de 1.98 palabras, la mediana es de 1 palabra y el total de palabras empleadas es

Representación semántica	# de elocuciones
característica	618
edad	264
ok	528
no	227
nombre	126
carta	43

Cuadro 3.2: Número de transcripciones por representación semántica en la parte de desarrollo del corpus Golem-UNIVERSUM.

Expectativa	Entrenamiento	Evaluación
característica	693	180
edad	211	54
ok	782	244
no	782	244
nombre	120	38
carta	693	180

Cuadro 3.3: Distribución de las expectativas en el corpus Golem-UNIVERSUM.

de 1588. Como se puede apreciar los valores varían, ya que el reconocedor de voz comete errores al detectar la secuencia de palabras.

La distribución del tipo de representación semántica que se tiene asignada a cada elocución en la parte de entrenamiento se muestra en el cuadro 3.2.

Las expectativas de las transcripciones se encuentran distribuidas como se muestra en el cuadro 3.3. Por lo observado en este cuadro, se puede notar que las transcripciones pueden estar relacionadas a más de una expectativa.

El uso del corpus, tanto en su parte de desarrollo como de evaluación, se basa no en el número de transcripciones, sino en el número de usuarios. El número total de usuarios con los que cuenta la parte de desarrollo es de 132, con un promedio de 13.68 transcripciones por usuario.

Representación semántica	# de elocuciones
característica	163
edad	60
ok	184
no	53
nombre	38
carta	18

Cuadro 3.4: Número de transcripciones por representación semántica en la parte de evaluación del corpus Golem-UNIVERSUM.

### Corpus de evaluación

El número total de elocuciones con las que cuenta el corpus en la parte de evaluación es de 516. Para el caso de las transcripciones manuales, el promedio de la longitud es de 1.87 palabras y de 2 palabras para la mediana. Este tipo de transcripciones emplean un total de 966 palabras. En lo que respecta a las transcripciones automáticas, el promedio de la longitud es de 1.97 palabras, la mediana es de 1 palabra y el total de palabras empleadas es de 1021. Las expectativas asociadas a las transcripciones para la parte de evaluación se encuentran distribuidas como se muestra en el cuadro 3.3.

El número total de usuarios con los que cuenta la parte de evaluación es de 40, con un promedio de 12.9 transcripciones por usuario.

La distribución del tipo de representación semántica que se tiene asignada a cada elocución en la parte de evaluación del corpus se muestra en el cuadro 3.4.

## 3.2. Lógica de Markov

El desarrollo e implementación del pársers semántico que se propone en este trabajo, se basa en el modelado de las relaciones entre la representación semánti-

ca y la transcripción de la elocución de entrada. En este trabajo el modelado de estas relaciones se realiza por medio de Lógica de Markov. Esta lógica se aplica considerando el problema del procesamiento semántico como una tarea de clasificación dentro del contexto de los métodos discriminativos. Es decir, que la representación semántica obtenida por medio de este modelado deberá estar basado en observaciones realizadas previamente. En el marco de este trabajo, la salida del procesamiento semántico consiste en una representación semántica, y las palabras que forman parte de las elocuciones de entrada se perciben como observaciones. Lógica de Markov es empleada para definir e instanciar redes lógicas de Markov.

La lógica de Markov se puede definir como:

$$\{(\phi_i, w_i)\} \quad (3.1)$$

Donde  $\phi_i$  representa una fórmula en lógica de primer orden y  $w_i$  representa un número real, que puede tener valores positivos o negativos. Las redes lógicas de Markov describen un conjunto de posibles mundos para los cuales la fórmula es satisfecha. Estas redes permiten manejar cierta incertidumbre, ya que en lugar de clasificar todos los mundos como consistentes o inconsistentes, les asigna un peso. El efecto de esta asignación es aumentar la flexibilidad de las fórmulas.

### 3.2.1. Redes Lógicas de Markov

Las redes de Markov son la base sobre la cual se desarrolla la teoría de la Lógica de Markov. Para definir una red lógica de Markov se requiere de un vocabulario formado de constantes y predicados. Las primeras hacen referencia a los objetos del dominio, mientras que las segundas hacen referencias a las relaciones entre los

objetos o constantes. Para este trabajo los objetos del dominio pueden ser, por ejemplo, la ortografía de una palabra (*rojo, si, años*) o su posición (*1, 2, 3*) dentro de la elocución. En cuanto a los predicados, estos son utilizados para definir relaciones entre los objetos. Utilizando las constantes anteriores se puede definir el predicado *palabra/2* que representa la relación entre la ortografía y la posición. La representación de este predicado utilizando los ejemplos de ortografía y posición sería *palabra(rojo,1)*, *palabra(si,2)* y *palabra(años,3)*. Empleando el predicado *palabra/2*, se puede definir un mundo con el cual se representa la elocución *se trata de un planeta*:

$$palabra(se,1), palabra(trata,2), palabra(de,3), palabra(un,4), palabra(planeta,5)$$

Otros objetos que se pueden tener dentro del dominio son las características (*edad, si, nombre*) o los valores que pueden tomar estas (*14, pablo, planeta*).

Otro predicado que se puede construir empleando los objetos característica y valor es *representacion/2*. Para la representación semántica *caracteristica(planeta)* se tiene:

$$representacion(caracteristica, planeta)$$

o para el caso de la representación *nombre(pablo)* se tiene el predicado:

$$representacion(nombre, pablo)$$

Empleando las constantes y los predicados, se puede definir una fórmula en lógica de primer orden con la que se describe las relaciones estructurales de un problema en particular. Para el caso de este trabajo la fórmula:

$$\phi_1 : palabra(p_1, o_1) \wedge representacion(caracteristica, valor)$$

define la relación entre la palabra  $o_1$  en la posición  $p_1$  con la representación semántica denotada por el predicado  $representacion(caracteristica, valor)$ . A esta relación se le pueden asignar pesos dependiendo del par indicado por palabra-representación para indicar la preferencia de algunos pares sobre otros. Lo cual implica que existen algunas palabras con sus respectivas posiciones que se presentan una mayor cantidad de veces en comparación con otras.

### 3.2.2. Lógica de Markov como modelo gráfico

Una red lógica de Markov puede representarse de manera gráfica por medio de un grafo de factores como el que se describe en el capítulo anterior para los CRF's. Tomando en cuenta lo anterior, la fórmula:

$$\phi_2 : palabra(1, planeta) \wedge palabra(caracteristica, planeta)$$

que es la misma mostrada en  $\phi_1$  solo que con valores en lugar de las constantes empleadas en el dominio, puede ser mostrada de manera gráfica como se muestra en la Figura 3.1. Existen sólo dos nodos en el grafo, uno para cada predicado (palabra, representación), conectados por el factor  $\phi_1$  que corresponde a la fórmula en lógica de primer orden.

Como se mencionó anteriormente, Lógica de Markov sirve para instanciar redes lógicas de Markov empleando lógica de primer orden. Para el caso de la fórmula

$$\phi_3 : palabra(p, o) \wedge representacion(caracteristica, planeta)$$

con la elocución de entrada *se trata de un planeta* genera una secuencia repetitiva de la relación entre cada una de las palabras de la elocución, tal y como se muestra en la Figura 3.2.



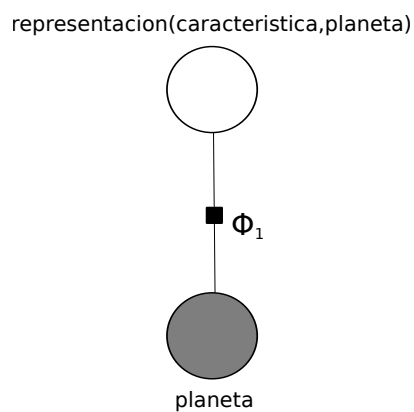


Figura 3.1: Grafo de factores para un fórmula.

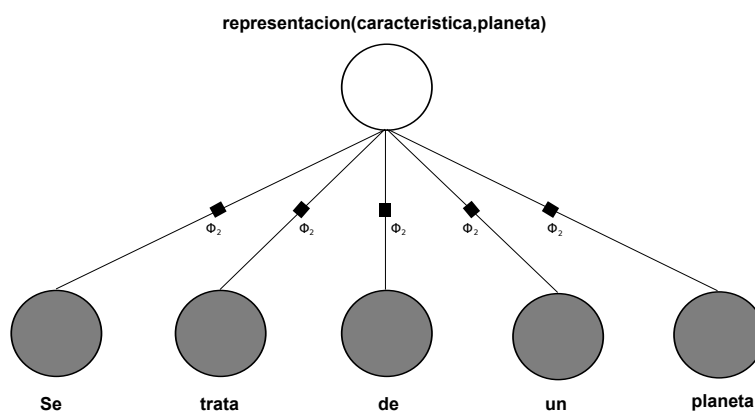


Figura 3.2: Grafo de factores para un posible mundo.

Al modelo que se presenta en  $\phi_1$ , en lo que resta de este documento se le llamará modelo base.

Lógica de Markov es un método discriminativo, es decir, el resultado obtenido se basa en observaciones realizadas previamente. Al proceso de realizar observaciones sobre el corpus se le conoce como entrenamiento. En Lógica de Markov el entrenamiento se lleva a cabo mediante el algoritmo conocido como MIRA (por sus siglas en inglés, Margin Infused Relaxed Algorithm). Este algoritmo tiene la

característica de ser *online* y discriminativo. El hecho de que sea *online* implica que la entrada la va analizando por partes. En el caso del corpus Golem-UNIVERSUM analiza las fórmulas de primer orden relacionadas a cada transcripción por separado para ir modificando los pesos asignados. La modificación de los pesos se hace empleando factores para no hacer los cambios drásticos. Los factores empleados indican que el algoritmo MIRA es discriminativo (McDonald et al., 2005). Para obtener la representación semántica de una transcripción de entrada se emplea el algoritmo CPI (por sus siglas en inglés, Cutting Plane Inference). Para encontrar la representación este algoritmo instancia solo partes de la red Lógica de Markov que puede proporcionar la solución, disminuyendo el tiempo que puede tomar realizar esta tarea (Riedel, 2008).

### 3.3. Métricas de evaluación

Con la finalidad de determinar cuál es el desempeño del párser semántico, tanto como el que se encuentra implementado actualmente en el sistema Golem-UNIVERSUM como el obtenido en este trabajo, se proponen diferentes métricas de evaluación. A continuación se explican cuáles son cada una de ellas y en qué consisten, además de la razón por las que fueron escogidas.

La primer métrica es la exactitud, la cual está definida por la siguiente fórmula:

$$Exactitud = \frac{\# \text{ representaciones semánticas asignadas correctamente}}{\# \text{ total de elocuciones}} \quad (3.2)$$

Como se puede apreciar en la fórmula 3.2 esta métrica muestra de una mane-

ra estricta el desempeño del párser semántico ya que considera sólo la cantidad de representaciones semánticas asignadas correctamente. Este aspecto es de suma importancia porque dentro del funcionamiento del párser en el sistema Golem-UNIVERSUM son sólo las representaciones asignadas correctamente las que indican adecuadamente cuál fue la intención original del usuario al momento de emitir la elocución, y son únicamente estas representaciones las que aseguran una respuesta congruente del sistema al usuario.

En este punto cabe señalar la diferencia entre la exactitud y las otras métricas empleadas. Mientras que la exactitud observa a la representación como un todo, siendo ésta la base de su rigidez, las otras métricas son capaces de observar por partes a la representación provocando que sean menos estrictas al ser aplicadas. La idea sobre la que parten las otras métricas son:

- Comparar la cantidad de construcciones correctas de partes de la representación contra la cantidad de veces que se falló (Zettlemyer and Collins, 2007).

En ambas opciones, se debe considerar las mediciones que se obtienen de la precisión y la cobertura. La primera medición se refiere a la dispersión de los valores obtenidos de repetidas mediciones de una magnitud (Olson and Delen, 2008). En otras palabra es el número de resultados correctos dividido entre el número de todos los resultados obtenidos. Para este caso se considera que mientras más bajo sea el valor de la dispersión, se considera que las mediciones obtenidas son más exactas. A continuación se muestra la fórmula 3.3 utilizada para medir la precisión de la construcción de representaciones semánticas completas.

$$Precisión = \frac{\# \text{ total de positivos verdaderos}}{\# \text{ total de positivos verdaderos} + \# \text{ total de falsos positivos}} \quad (3.3)$$

Donde un positivo verdadero significa que se obtuvo la representación semántica que se esperaba por medio del párser semántico, mientras que un falso positivo indica que la representación semántica es construida cuando en realidad no se debió de obtener nada. Como se mencionó en la sección anterior la representación semántica se encuentra dividida en dos partes: característica y valor. Para cada una de estas partes se puede obtener los positivos verdaderos y los falsos positivos.

La precisión se puede obtener de manera local y de manera global. Para el caso del cálculo global se emplea un registro del total de positivos verdaderos y falsos positivos para al final realizar un solo cálculo con estos valores. Para la precisión local, el cálculo se realiza para cada una de las representaciones semánticas, se guardan los resultados y es un promedio de estos resultados el que se reporta.

La cobertura se refiere a la diferencia que existe entre el valor medido y el valor real de una medición. En otras palabras la cobertura es el número de resultados correctos dividido entre el número de resultados que se debieron de haber obtenido (Olson and Delen, 2008). En seguida se muestra la fórmula 3.4 utilizada para medir la cobertura de la construcción de representaciones semánticas completas.

$$Cobertura = \frac{\# \text{ total de positivos verdaderos}}{\# \text{ total de positivos verdaderos} + \# \text{ total de falsos negativos}} \quad (3.4)$$

Donde un falso negativo indica que no se obtuvo una representación semántica,

a pesar de que el pársers sí la debió obtener.

Como en el caso de la precisión, la cobertura también se puede obtener de manera local y de manera global. Para el caso del cálculo global se emplea un registro del total de positivos verdaderos y falsos negativos para finalmente realizar un solo cálculo con estos valores. Para la cobertura local, el cálculo se realiza para cada una de las representaciones semánticas, nuevamente se guardan los resultados y lo que se reporta es un promedio de estos resultados.

Para relacionar las dos mediciones anteriores, precisión y cobertura, se utiliza el valor de f-score. Este valor es considerado como el promedio de la precisión y la cobertura. El rango de valores que f-score puede tomar van de 0 a 1, donde 0 indica que se tiene el peor desempeño y 1 indica que se encontró con el mejor desempeño. La fórmula 3.5 muestra como calcular el valor de f-score.

$$f - score = 2 * \frac{precisión * cobertura}{precisión + cobertura} \quad (3.5)$$

El f-score puede ser local o global. Esto último depende de los valores de la cobertura y la precisión que se utilice.

En resumen, los resultados que se obtengan durante los experimentos definidos para este trabajo se mostrarán en estas tres métricas (exactitud, f-score global, f-score local), sin embargo se pondrá énfasis en la exactitud al ser más estricta como se explicó anteriormente.

El contenido de este capítulo nos presentó una definición de Lógica de Markov y sus características más importantes. Además se presentaron las métricas que serán de utilidad para evaluar los resultados proporcionados por los experimentos realizados y plantear una comparación. En el capítulo siguiente se detallan los

experimentos realizados así como los resultados de los mismos.

## Capítulo 4

# Procesamiento semántico sin contexto

El contenido de este capítulo presenta los experimentos realizados para determinar el modelo en Lógica de Markov que mejor realiza el procesamiento en el sistema Golem-UNIVERSUM. Los experimentos descritos en este capítulo tienen como característica no incluir el contexto de la transcripción de la elocución durante el desarrollo de los mismos. Como se mencionó en el capítulo 2, dentro de este trabajo se entiende al contexto de la elocución como la expectativa que tiene el sistema al momento de esperar una elocución de entrada.

El contenido de este capítulo se encuentra estructurado como se describe a continuación. En la sección 4.1 se describen los experimentos que se realizan empleando las transcripciones manuales del corpus Golem-UNIVERSUM, así como los resultados obtenidos. En la sección 4.2 se explican los experimentos realizados con las transcripciones automáticas de las elocuciones. En la sección 4.3 se presenta

una discusión de la importancia de los resultados obtenidos.

## 4.1. Experimentos con transcripciones manuales

Los experimentos que se describen en esta sección son realizados empleando tanto la parte de desarrollo como de evaluación del corpus. Las transcripciones manuales de la parte de desarrollo son empleadas para el experimento de baseline, el experimento parametrizador y los experimentos con los modelos extendidos. Por otra parte, para los experimentos de evaluación se emplea la parte del corpus con el mismo nombre, usando tanto transcripciones manuales como automáticas.

### 4.1.1. Baseline

En esta sección se explica el desarrollo del experimento que tiene como objetivo el obtener una evaluación del desempeño del pársers semántico que actualmente se encuentra instalado en el sistema Golem-UNIVERSUM. Este experimento será la base con la que se realizarán las comparaciones con los resultados de los demás experimentos. Esta evaluación servirá como referencia para determinar si la solución propuesta en este trabajo muestra un mejor desempeño del sistema en el momento de llevar a cabo el procesamiento semántico de las elocuciones de entrada.

Este experimento es de suma importancia, ya que si bien se ha comentado en el capítulo 1 que el pársers semántico implementado en el sistema Golem-UNIVERSUM presenta un desempeño deficiente, no se ha demostrado en qué medida se presenta esta deficiencia. También hay que considerar que hasta este punto no se tenían definidas las métricas de evaluación con las cuales se mediría



el desempeño, ya sea del párser semántico o de los modelos que se proponen.

Para obtener los valores del desempeño del procesamiento semántico en el sistema, se emplearon en su totalidad las transcripciones manuales de las elocuciones de entrada del corpus Golem-UNIVERSUM. Las transcripciones fueron sometidas al procesamiento semántico realizado por el párser del sistema. Para llevar a cabo este experimento, el párser fue modificado de manera que no fuera necesario el incluir la expectativa del sistema al momento de su ejecución. Con la finalidad de llevar a cabo las mediciones, las representaciones asignadas por el párser fueron comparadas con las representaciones asignadas manualmente a las transcripciones.

### **Resultados**

El valor de la exactitud, mostrado en el cuadro 4.1, es de utilidad para indicar que se tiene un porcentaje de error que afecta considerablemente el desempeño del párser durante la asignación de representaciones semánticas completas a las transcripciones de las elocuciones. El hecho de que falle dos representaciones de cada 10 provoca que el sistema repita preguntas en más de una ocasión durante las interacciones con los usuarios causando con esto que el juego se vuelva tedioso.

Como también se observa en el cuadro 4.1, las medidas f-score global y f-score local presentan valores relativamente más altos al de la exactitud, aunque esto no implica una mejora en el desempeño. Cabe aclarar en ese punto que la asignación parcial correcta de una representación semántica tampoco es de mucha utilidad para el sistema Golem-UNIVERSUM, ya que no se tiene contempladas situaciones donde solo una parte de la representación es correcta. Esto prueba que la exactitud es más estricta que los valores de f-score global o local.

Medida	Valor
exactitud	75.80 %
f-score local	78.79 %
f-score global	75.85 %

Cuadro 4.1: Resultados de la evaluación del desempeño del párser semántico del sistema Golem-UNIVERSUM con transcripciones manuales.

#### 4.1.2. Experimento parametrizador

Es en este punto donde se empieza a trabajar con modelos en Lógica de Markov. Para el entrenamiento de los modelos se pueden modificar algunos parámetros que nos ayudan a tener un mejor desempeño en algunos casos, mientras que se presentan fallas en otros. Con la finalidad de identificar cuáles son los parámetros que proporcionan un mejor entrenamiento de los modelos es por lo que se propone este experimento. El modelo base, como se le llamó en el capítulo anterior, es empleado durante estos experimentos.

Pese a que son varios los parámetros que se pueden modificar, el objetivo de estos experimentos es determinar cuál es la cantidad de épocas con las que se garantiza un buen entrenamiento del modelo. Las épocas son la cantidad de veces que el corpus será analizado en su totalidad para el entrenamiento de los modelos, lo que implica que entre mayor sea el número de épocas mayor es el tiempo que se requiere para que el modelo sea entrenado. El número de épocas con las cuales se realizan estos entrenamientos está en el rango de 1 a 20. Para cada uno de estos valores se trabaja con la parte de entrenamiento del corpus Golem-UNIVERSUM de la manera en que se muestra en la Figura 4.1. El corpus se divide en 10 partes iguales, para la primera iteración se toma la parte 1 para realizar una evaluación del modelo entrenado con las 9 partes restantes. Para la segunda iteración la parte

1 se integra para el entrenamiento y la parte 2 se toma para la evaluación. Así sucesivamente hasta terminar con todas la partes. Al final se tiene todo el corpus etiquetado que nos sirve para reportar el desempeño de este modelo. A esta forma de trabajar se conoce como validación cruzada.

## Corpus Golem-UNIVERSUM

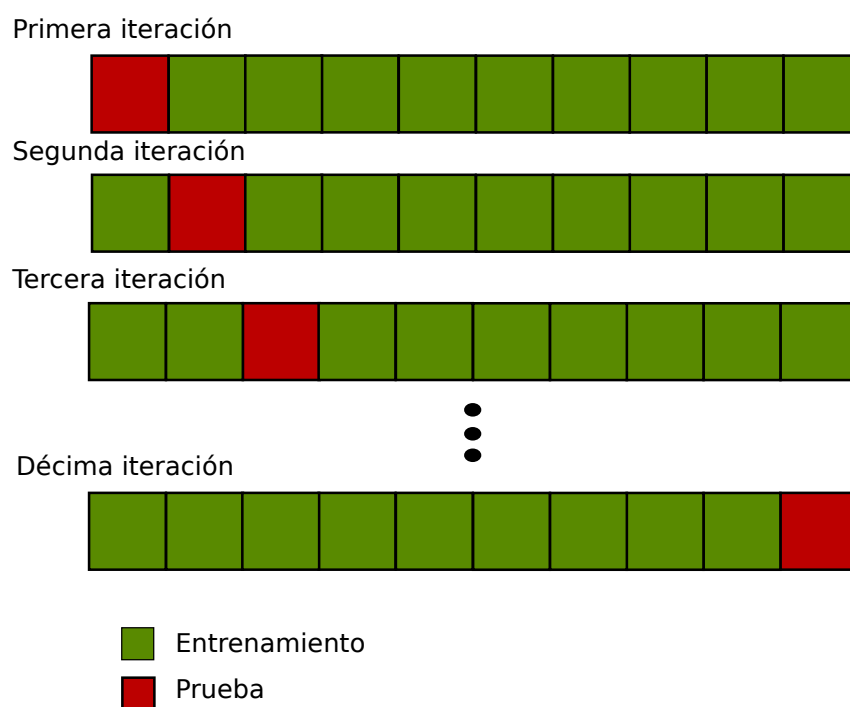


Figura 4.1: Esquema de validación cruzada para el corpus Golem-UNIVERSUM.

### Resultados

Los resultados que se obtienen para cada una de las épocas con las que se llevó a cabo este experimento se muestran en la Figura 4.2. En la gráfica se muestra que la línea marcada con rombos indica el valor de exactitud del experimento baseline, mientras que la línea marcada con cuadrados presenta el mismo valor

para el modelo base.

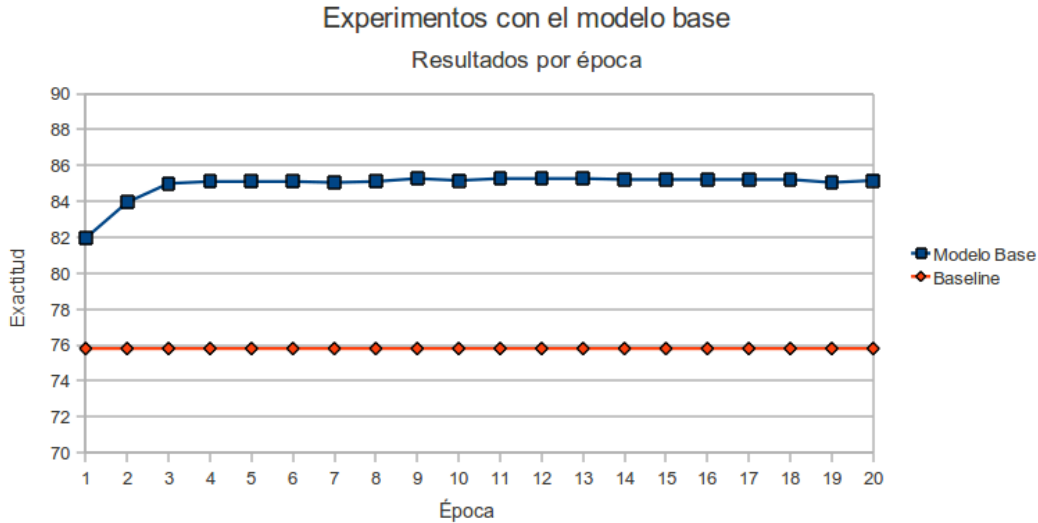


Figura 4.2: Curva de aprendizaje de la exactitud para los experimentos con el modelo base.

Además en la Figura 4.2 se observa la curva de aprendizaje para la exactitud, la cual mantiene un aumento en el aprendizaje dentro del rango de 1 a 9 para el modelo base. De este último valor hasta el 20, el aprendizaje presenta un comportamiento constante. De la gráfica también se puede observar que la época que representa un mejor comportamiento del modelo es la número 9 ya que presenta el máximo valor en el desempeño. Este valor obtenido es de suma importancia para los experimentos siguientes ya que indica la época en la que los modelos deben ser entrenados para obtener un buen desempeño. Los valores obtenidos en la época 9 para las métricas de evaluación se muestran en el cuadro 4.2.

En cuanto a la comparación directa del desempeño del modelo base contra la ejecución del pársers del sistema Golem-UNIVERSUM se puede inferir que el modelo trabaja mejor en una medida del 5% . Esto se puede traducir en un mejor

Modelo	Modelo base
Época	9
exactitud	85.27 %
f-score local	91.00 %
f-score global	89.07 %

Cuadro 4.2: Valores del desempeño del modelo base en la época que presenta el mejor aprendizaje.

desempeño del sistema conversacional, ya que es menor la cantidad de errores al asignar representaciones semánticas lo que implica una interacción más fluida entre el sistema y el usuario.

#### 4.1.3. Modelos extendidos

Hasta este momento se ha trabajado con el modelo base en Lógica de Markov, para el que sólo se requieren las transcripciones de las elocuciones y las representaciones semánticas asignadas a cada una de ellas. Pero la interrogante que surge en este punto es si se puede extender el modelo y cuál sería el efecto de la extensión. La extensión de los modelos en Lógica de Markov se puede presentar de dos maneras:

- Crear estructuras más complejas del modelo mediante el empleo de la información que se tiene hasta este momento.
- Agregando información nueva directamente al modelo.

En seguida se analiza cada una de las opciones citadas. En el primer caso, para poder complicar la estructura del modelo se requiere de la definición de nuevas fórmulas de primer orden que relacionen a más de dos predicados. En esta situación no se presenta la restricción que sean distintos los predicados. Este tipo de extensión

se puede aplicar al modelo base, el resultado que se obtiene se muestra en la Figura 4.3.

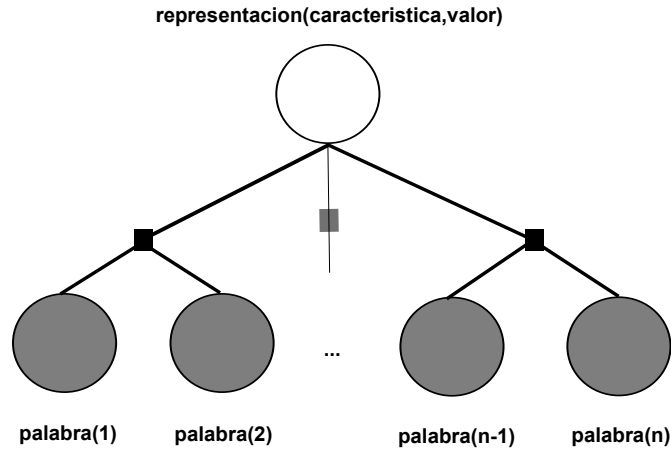


Figura 4.3: Modelo base extendido mediante la creación de estructuras complejas.

Como se aprecia en el diagrama, se ha incrementado la complejidad de la estructura del modelo al organizar las palabras en bigramas. Aunque en análisis de texto se pueden realizar grupo de sílabas o letras, para este caso un bigrama es un grupo de dos palabras. La fórmula en lógica de primer orden que nos sirve para relacionar los predicados es:

$$\phi_4 : palabra(p_1, o_1) \wedge palabra(p_1 + 1, o_2) \wedge representacion(caracteristica, valor)$$

La otra forma de extender los modelos en Lógica de Markov es agregando información nueva. Para una mejor comprensión de esta forma de extensión se presenta la Figura 4.4, donde se muestra el modelo base al que se le adicionó información. En el grafo mostrado se puede notar que la forma de agregar información es agregar variables observables al modelos. Para este modelo se agrega la etiqueta que muestra la categoría gramatical de cada palabra. La fórmula de primer orden para

este modelo es

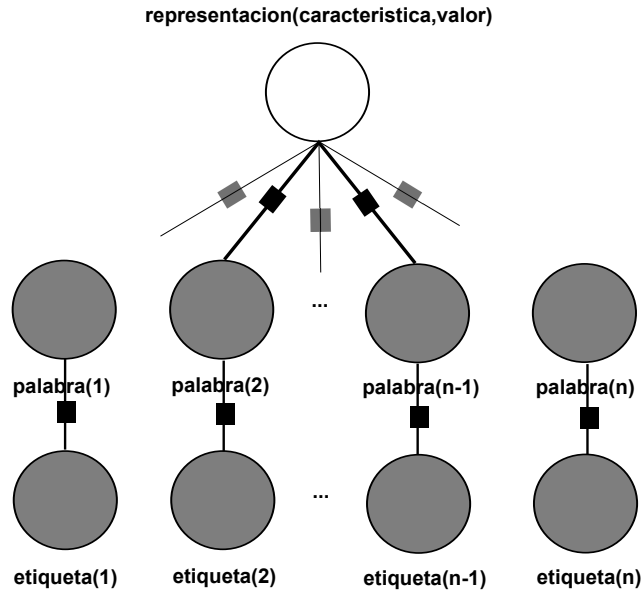


Figura 4.4: Modelo base extendido mediante la adición de información nueva.

$$\phi_5 : palabra(p_1, o_1) \wedge etiqueta(p_1, e_1) \wedge representacion(caracteristica, valor)$$

donde  $etiqueta(p_1, e_1)$  es el predicado que se definió para relacionar al objeto etiqueta ( $e_1$ ) con la posición de la palabra ( $p_1$ ) a la que le corresponde. En el cuadro 4.3 se muestran ejemplos de las etiquetas para las palabras de las transcripciones, tanto manuales como automáticas, mostradas en el cuadro 3.1.

El modelo base y los modelos extendidos descritos fueron combinados de manera que se generó una serie de modelos que se listan a continuación:

- Base + Bigramas.

---

*parece un ojo*  
VMIP3S0 DI0MS0 I  
*grande de un ojo*  
AQ0CS0 NCFS000 DI0MS0 I

*tengo 14 años*  
VMIP1S0 DUMMY NCMP000  
en en este tengo catorce  
VSIP3S0 SPS00 DD0MS0 VMIP1S0 DUMMY

*mi nombre es erik*  
DP1CSS VMM03S0 VSIP3S0 DUMMY  
*alambres erico*  
VMSP2S0 DUMMY

*si claro que si*  
CS AQ0MS0 CS CS  
*claro de lo es*  
AQ0MS0 NCFS000 DA0MP0 VSIP3S0

---

Cuadro 4.3: Transcripciones de elocuciones del corpus Golem-UNIVERSUM etiquetadas con categorías gramaticales.

- Base + Etiquetado.
- Base + Bigramas + Etiquetado.

En los modelos donde se emplea el etiquetado se manejaron 4 diferentes posiciones de las palabras que son:

- Etiqueta de la palabra actual.
- Etiqueta de la palabra actual + Etiqueta de la palabra anterior.
- Etiqueta de la palabra actual + Etiqueta de la palabra siguiente.
- Etiqueta de la palabra actual + Etiqueta de la palabra anterior + Etiqueta de la palabra siguiente.

Las consideraciones anteriores dan un total de 9 modelos extendidos. Cada uno de estos modelos fue sometido a un entrenamiento con el corpus de desarrollo en



un rango de épocas de 7 a 11. Este rango se determinó al considerar que el modelo base refleja su mayor aprendizaje en la época 9, que es el centro del rango que se empleó.

### Resultados

Al realizar la recolección de los datos que arrojaron los 9 modelos, se descubrió que todos mostraban una curva de aprendizaje constante. Los resultados de estos experimentos pueden ser consultados en el apéndice A. Por esta razón se decidió tomar los modelos que en esas curvas presentaban el mejor desempeño y someterlos a un entrenamiento similar al del experimento parametrizador.

Los modelos que se escogieron fueron:

- Base + Bigramas.
- Base + Bigramas + Etiqueta de la palabra actual + Etiqueta de la palabra siguiente.

Para una mejor identificación de estos modelos, en el resto de este documento, al primero de ellos se le llamará modelo bigramas y al segundo modelo etiquetado. En la Figura 4.5 se presenta la curva de aprendizaje del modelo bigramas para la métrica de la exactitud, mientras que en la Figura 4.6 se observa la curva de aprendizaje para el modelo etiquetado. En ambas gráficas se emplea el desempeño del modelo base como referencia.

En el caso del modelo bigramas, se puede observar una mayor rapidez en alcanzar su máximo aprendizaje que se presenta en la época 4; para el caso del modelo etiquetado es en la época 5. A pesar de que en ambos se alcanza con una mayor rapidez el máximo aprendizaje, ninguno de los dos modelos presenta un mejor desempeño en comparación con el modelo base. Esto se comprueba con la infor-

Modelo	Modelo base	Modelo Bigramas	Modelo Etiquetado
Época	9	4	5
exactitud	85.27 %	84.71 %	84.77 %
f-score local	91.00 %	90.47 %	90.44 %
f-score global	89.07 %	88.46 %	88.34 %

Cuadro 4.4: Comparación del desempeño de los modelos en las épocas donde presentan un mejor aprendizaje con transcripciones manuales.

mación del cuadro 4.4 que muestra los valores de exactitud, f-score local y f-score global para cada modelo.

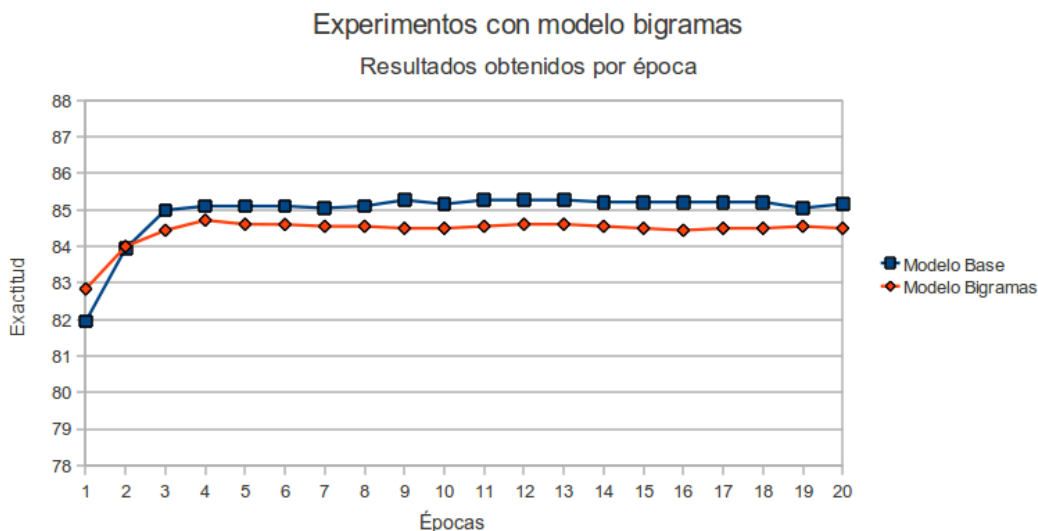


Figura 4.5: Curva de aprendizaje para los experimentos con el modelo bigramas.

A pesar de que los modelos extendidos no superan al modelo base, en lo que resta de este trabajo, se seguirá trabajando con los tres modelos, ya que presentan un mayor desempeño que el baseline que se tiene. Hasta este punto se considera que la implementación de cualquiera de estos modelos reflejaría un mejor entendimiento del lenguaje en el sistema Golem-UNIVERUSM.

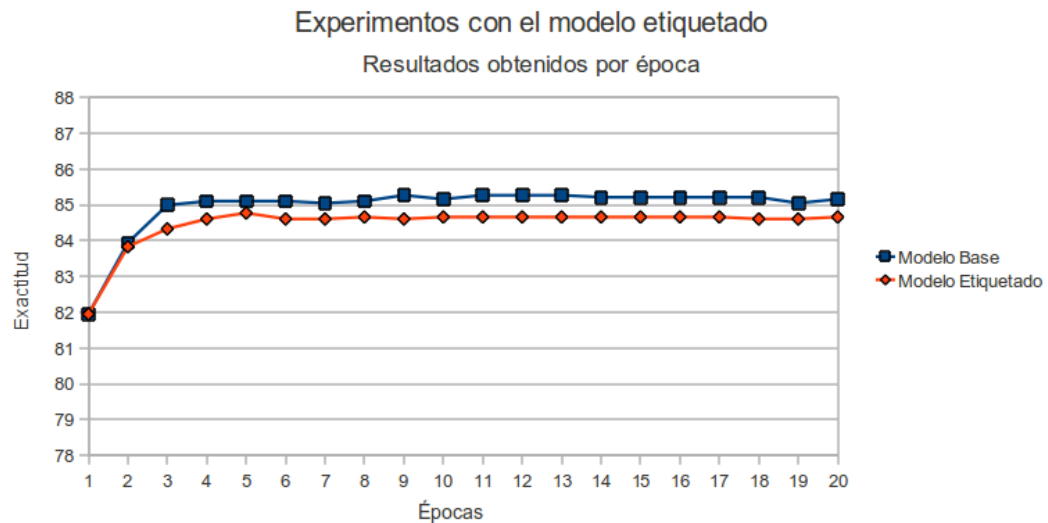


Figura 4.6: Curva de aprendizaje para los experimentos con el modelo etiquetado.

#### 4.1.4. Evaluación

Con la finalidad de medir la mejoría que se obtendría al implementar cada uno de los modelos en Lógica de Markov en el sistema Golem-UNIVERSUM se propone la realización de evaluaciones del desempeño de los modelos. El entrenamiento de los modelos se realiza empleando la totalidad de transcripciones manuales de la parte de desarrollo del corpus Golem-UNIVERSUM. Para el entrenamiento de cada modelo se tomarán en cuenta las épocas que se muestran en el cuadro 4.4. Para la evaluación del desempeño de los modelos entrenados se emplean las transcripciones, tanto manuales como automáticas, de la parte de evaluación del corpus Golem-UNIVERSUM.

La primera evaluación se realiza con las transcripciones manuales, esto con el propósito de mantener una consistencia en el tipo de datos que se manejan tanto en el entrenamiento como en la evaluación. Los resultados de esta evaluación se

Modelo	Baseline	Modelo Base	Modelo Bigramas	Modelo Etiquetado
Época	-	9	4	5
exactitud	74.80 %	86.95 %	85.65 %	84.88 %
f-score local	78.87 %	92.60 %	94.57 %	90.31 %
f-score global	74.67 %	91.105 %	89.59 %	88.06 %

Cuadro 4.5: Comparación del desempeño de los modelos en las épocas donde presentan un mejor aprendizaje evaluados con transcripciones manuales.

presentan en la gráfica de la Figura 4.7.

El baseline que se muestra en esta gráfica se obtuvo al someter a un procesamiento semántico todas las transcripciones manuales de la parte de evaluación del corpus, sin incluir las expectativas.

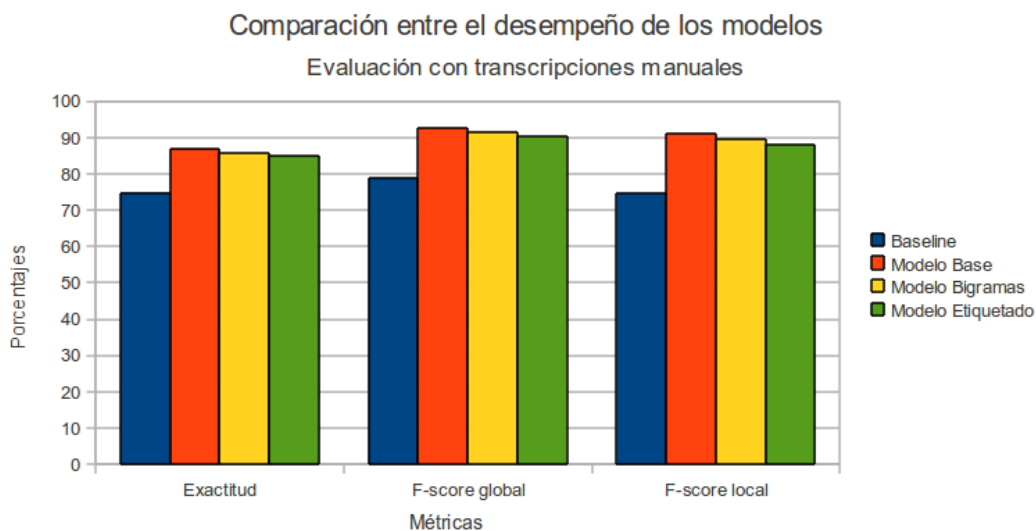


Figura 4.7: Resultados de la evaluación de los modelos entrenados con transcripciones manuales empleando transcripciones del mismo tipo.

La segunda de las evaluaciones se realiza con las transcripciones automáticas. Este tipo de transcripciones son con las que trabaja el párser semántico durante

Modelo	Baseline	Modelo Base	Modelo Bigramas	Modelo Etiquetado
Época	-	9	4	5
exactitud	45.93 %	48.06 %	45.15 %	45.73 %
f-score local	49.90 %	55.91 %	53.97 %	55.52 %
f-score global	45.04 %	53.51 %	51.48 %	53.23 %

Cuadro 4.6: Comparación del desempeño de los modelos en las épocas donde presentan un mejor aprendizaje evaluados con transcripciones automáticas.

las interacciones lo cual, a su vez, permite recrear un ambiente similar a tener al modelo trabajando en el sistema Golem-UNIVERSUM directamente. En la Figura 4.8 se presentan los resultados de esta evaluación.

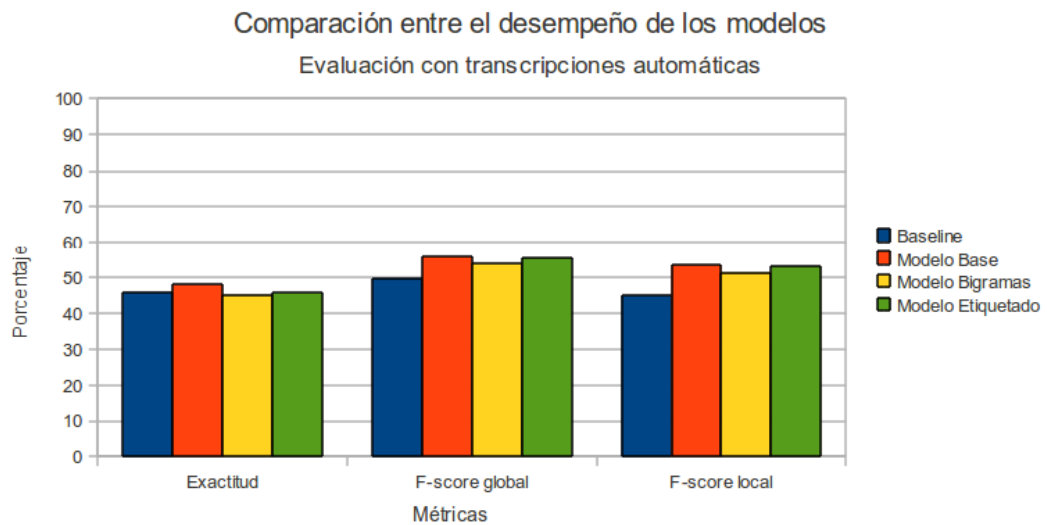


Figura 4.8: Resultados de la evaluación de los modelos entrenados con transcripciones manuales empleando transcripciones automáticas.

Las transcripciones automáticas de la parte de evaluación del corpus fueron sometidas a un procesamiento semántico en el párser que fue modificado para el experimento de baseline descrito anteriormente.

## 4.2. Experimentos con transcripciones automáticas

Es importante realizar experimentos con las transcripciones automáticas del corpus, ya que son éstas las que se obtienen del reconocedor de voz, lo que implica que son más parecidas a las que se presentan durante las interacciones del sistema con los usuarios. Para la realización de los experimentos que se describen en esta sección se emplea tanto la parte de entrenamiento como la de evaluación del corpus. La parte de entrenamiento se emplea para el experimento de baseline y los experimentos parametrizadores. Para los experimentos de evaluación se emplea la parte del corpus con el mismo nombre. Para los experimentos descritos se emplean únicamente las transcripciones automáticas del corpus.

### 4.2.1. Baseline

Este experimento tiene como finalidad obtener los valores del desempeño del procesamiento semántico en el sistema Golem-UNIVERSUM cuando se emplean transcripciones automáticas. En este experimento también se descartan las expectativas relacionadas a cada transcripción. Las transcripciones fueron sometidas al procesamiento semántico realizado por el pársers que se utilizó para el experimento baseline de las transcripciones manuales. Para llevar a cabo las mediciones, las representaciones obtenidas fueron comparadas con las representaciones asignadas manualmente a las transcripciones automáticas. Las mediciones obtenidas de los resultados de este experimento se muestran en el cuadro 4.7.

Como se puede observar el desempeño que se obtuvo es aún más deficiente que el del experimento con transcripciones manuales. La principal razón a la que se debe esto es que las transcripciones automáticas a veces ni siquiera contienen las

Medida	Valor
exactitud	39.29 %
f-score local	43.58 %
f-score global	39.80 %

Cuadro 4.7: Resultados de la evaluación del desempeño del párser semántico del sistema Golem-UNIVERSUM con transcripciones automáticas.

palabras clave que son necesarias para la técnica de asignación de representaciones implementada en el párser, mientras que las transcripciones manuales contienen, en la mayoría de los casos, palabras clave.

#### 4.2.2. Experimentos parametrizadores

Para los experimentos descritos en esta sección se emplearon 3 modelos: el base, el modelo bigramas y el modelo etiquetado. Estos modelos fueron elegidos por que presentan el mejor desempeño durante el procesamiento semántico de transcripciones manuales. La finalidad con la que estos experimentos son realizados es conocer cuál es el desempeño de estos modelos empleando transcripciones automáticas para su entrenamiento. Para cada modelo se realiza una serie de entrenamientos, durante los cuales se varía el número de épocas en un rango de 1 a 20. Se emplean las transcripciones automáticas de la parte de entrenamiento del corpus Golem-UNIVERSUM, sometiéndolo a un proceso de validación cruzada como el que se muestra en la Figura 4.1.

##### Resultados

En seguida se muestran los resultados de cada uno de los experimentos realizados, además se realiza una breve descripción de los mismos así como sus aportaciones a este trabajo.

La curva de aprendizaje para el modelo base se muestra en la Figura 4.9. Los valores arrojados por estos experimentos, si bien son superiores al baseline, no muestran un desempeño confiable durante el procesamiento semántico. Este modelo alcanza su desempeño máximo en la época 12, donde los valores indican que se tiene una razón de poco más de 5 representaciones asignadas correctamente de cada 10. Esto se traduciría en un desempeño considerablemente ineficiente durante las interacciones del sistema con el usuario.

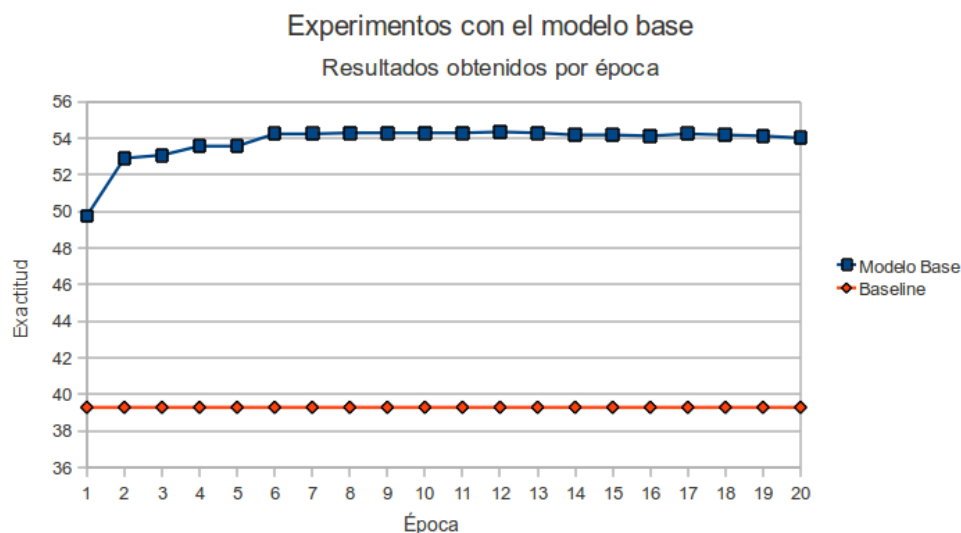


Figura 4.9: Curva de aprendizaje para el modelo base empleando transcripciones automáticas.

Los resultados de los experimentos realizados con los modelos bigramas y etiquetado se muestran en las Figuras 4.10 y 4.11 respectivamente. En ambas gráficas se puede observar que los resultados son comparados con los obtenidos con el modelo base. El análisis de estos datos presenta algunas similitudes. Ambos alcanzan su máximo aprendizaje en la época 8. De igual forma se puede notar un comportamiento similar de ambas curvas de aprendizaje y una semejanza en los valores



Modelo	Modelo base	Modelo Bigramas	Modelo Etiquetado
Época	12	8	8
exactitud	54.34 %	50.36 %	50.47 %
f-score local	61.69 %	59.67 %	59.54 %
f-score global	58.70 %	55.94 %	56.26 %

Cuadro 4.8: Comparación del desempeño de los modelos en las épocas donde presentan un mejor aprendizaje con transcripciones automáticas.

máximos alcanzados.

A pesar de que los resultados de estos dos modelos superan lo planteado en el experimento baseline, el desempeño no es mejor que el obtenido con el modelo base. Esto muestra una consistencia con los resultados logrados con transcripciones manuales donde se presenta la misma situación. En el cuadro 4.8 se presenta una comparación de los mejores resultados que se obtuvieron para cada modelo al ser entrenado con transcripciones manuales.

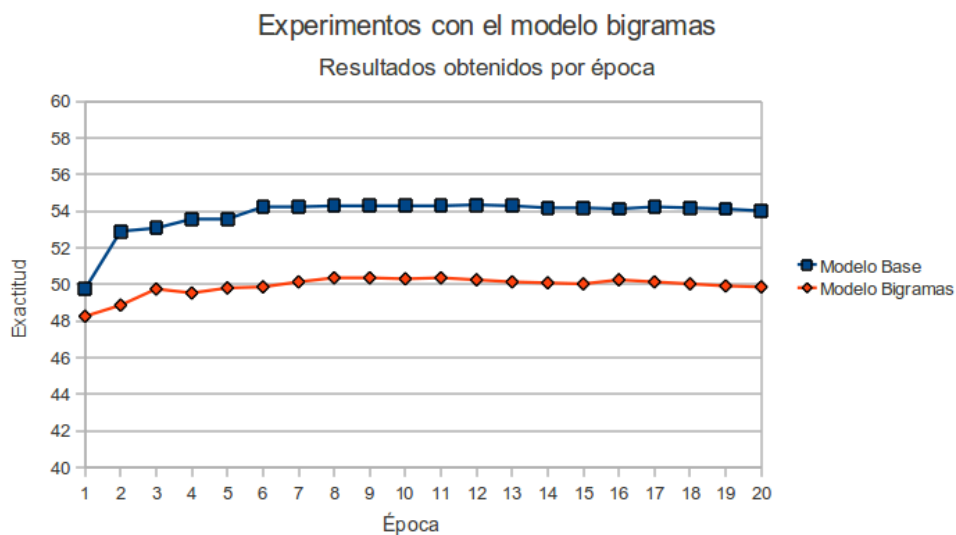


Figura 4.10: Curva de aprendizaje para el modelo bigramas empleando transcripciones automáticas.

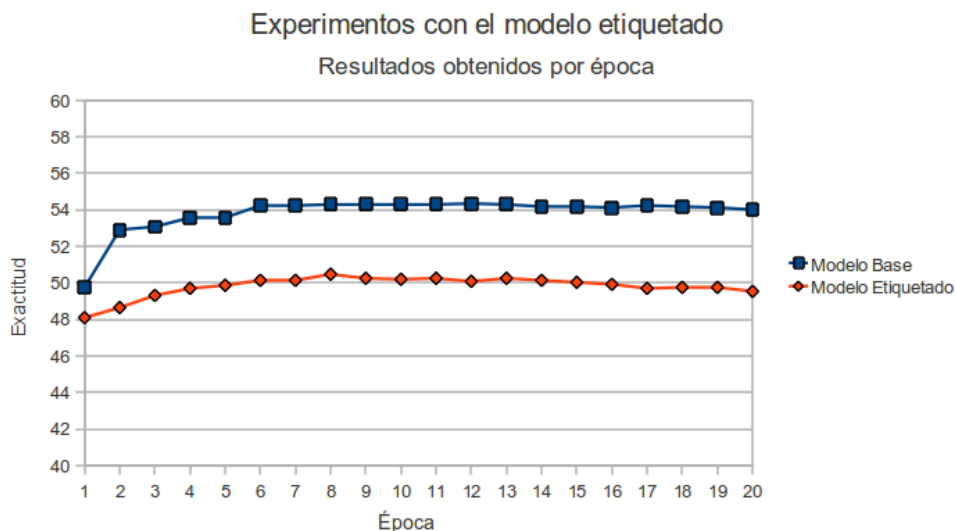


Figura 4.11: Curva de aprendizaje para el modelo etiquetado empleando transcripciones automáticas.

### 4.2.3. Evaluación

Como se mencionó anteriormente, se considera preciso conocer el desempeño de cada uno de los modelos al ser implementados en el sistema Golem-UNIVERSUM. Con esta finalidad se plantea la siguiente evaluación.

Para la evaluación que se desarrolla en esta sección, el aprendizaje del modelo se realiza empleando la totalidad de las transcripciones automáticas del corpus de entrenamiento. Para el entrenamiento de cada modelo se tomarán en cuenta las épocas que se muestran en el cuadro 4.8. Mientras que los resultados presentados se obtienen al evaluar estos modelos con las transcripciones automáticas del corpus de evaluación.

Los resultados de esta evaluación se presentan en la gráfica de la Figura 4.12. El baseline que se muestra en esta gráfica se obtuvo al someter a un procesamien-

to semántico todas las transcripciones automáticas de la parte de evaluación del corpus, sin incluir las expectativas.

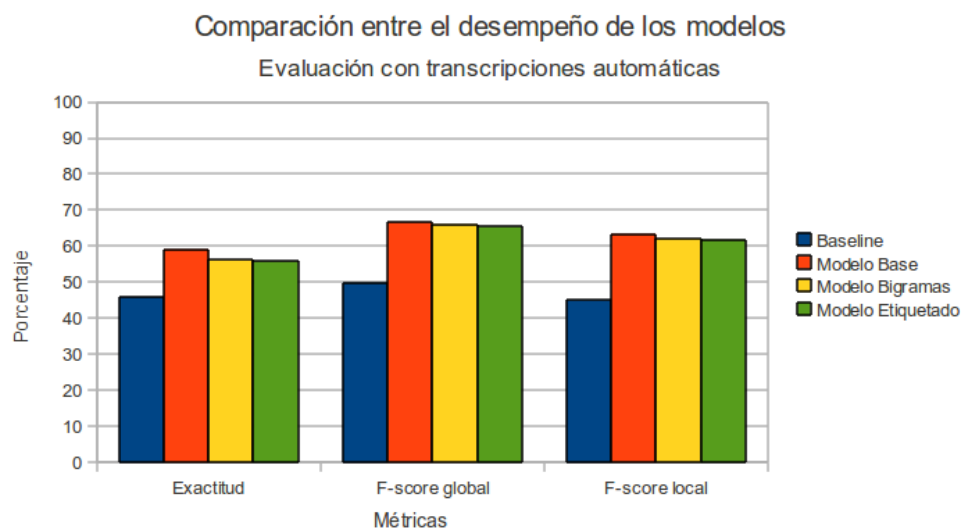


Figura 4.12: Resultados de la evaluación de los modelos empleando transcripciones automáticas.

Como se puede apreciar en la Figura 4.12, los resultados mantienen la misma tendencia que las evaluaciones realizadas con transcripciones manuales y automáticas realizadas anteriormente. El valor de baseline de esta evaluación nos indica que se asignan correctamente 4 representaciones semánticas de cada 10 elocuciones sometidas al procesamiento semántico. Los tres modelos empleados superan el valor obtenido ya que son capaces de asignar 5 representaciones de cada 10 elocuciones. Pero es el modelo base el que presenta el mejor desempeño. En el cuadro 4.9 se presentan los valores para las métricas de evaluación mostrados en la Figura 4.12.

Modelo	Baseline	Modelo Base	Modelo Bigramas	Modelo Etiquetado
Época	-	12	8	8
exactitud	45.93 %	59.11 %	56.20 %	55.81 %
f-score local	49.90 %	66.67 %	66.09 %	65.41 %
f-score global	45.04 %	63.41 %	62.06 %	61.79 %

Cuadro 4.9: Comparación del desempeño de los modelos en las épocas donde presentan un mejor aprendizaje con transcripciones automáticas.

### 4.3. Marco de los experimentos

Para el desarrollo de los experimentos descritos en este capítulo se utilizaron las siguientes herramientas:

- El control y la automatización de los experimentos se llevaron a cabo con *scripts* en lenguaje de programación Python.
- El etiquetado con categorías gramaticales se llevo a cabo en el programa FreeLing en su versión 2.2. Información acerca de la instalación y uso se puede encontrar en su página de internet<sup>1</sup>.
- El programa The Beast en su versión 0.0.2 se empleó para el entrenamiento y evaluación de los modelos en Lógica de Markov. Este programa implementa los algoritmos MIRA y CPI para entrenamiento y evaluación respectivamente.<sup>2</sup>

En lo referente a equipo se emplearon los listados a continuación:

- Computadora con un procesador Intel(R) Pentium(R) 4 CPU 3.06 GHz

<sup>1</sup><http://nlp.lsi.upc.edu/freeling/>

<sup>2</sup><http://code.google.com/p/thebeast/>

y 1GB en memoria RAM. El equipo cuenta con sistema operativo Ubuntu(Lucid) versión 10.04.

- Computadora con un procesador Intel(R) Core(TM) i3 CPU 3.07 GHz y 4GB en memoria RAM. El equipo cuenta con sistema operativo Ubuntu(Maverick) versión 10.10.

#### 4.4. Discusión

En este capítulo se mostraron los experimentos planteados con la finalidad de determinar el modelo en Lógica de Markov que llevará a cabo de la mejor manera el procesamiento semántico. Como resultado de estos experimentos se obtuvieron tres modelos (base, bigramas y etiquetado) que presentaron un mejor desempeño en comparación con el pársers implementado actualmente en el sistema Golem-UNIVERSUM. El modelo base presenta el mejor desempeño superando a los modelos bigramas y etiquetado. Esta tendencia se presenta en todas las evaluaciones realizadas, excepto en el caso donde emplean transcripciones automáticas para la evaluación de modelos entrenados con transcripciones manuales. La razón de que esto ocurra es la inconsistencia entre los datos empleados para el entrenamiento y evaluación.

En base a los resultados de las evaluaciones realizadas es prudente considerar que la implementación de cualquiera de los modelos se puede traducir en un mejor desempeño del pársers semántico, lo que a su vez produce interacciones más fluidas y coherentes entre el usuario y el sistema. En el caso del modelo base se debe considerar que el entrenamiento del modelo requiere de más tiempo que en el caso de los modelos bigramas y etiquetado, ya que se es necesario un mayor número de

épocas para asegurar un buen entrenamiento.

Por la foma en la que el sistema Golem-UNIVERSUM fue implementado, el corpus recolectado nos proporciona información que hasta este punto no se ha utilizado en los experimentos desarrollados. El siguiente capítulo presenta los experimentos que se realizan empleando la información del contexto de cada transcripción que hasta este punto se había descartado al momento de implementar los modelos en Lógica de Markov.

## Capítulo 5

# Procesamiento semántico con contexto

El objetivo que se tenía para este trabajo era mejorar el desempeño del pársers semántico del sistema Golem-UNIVERSUM. Hasta este momento se puede decir que este objetivo ha sido alcanzado, al considerar los resultados de los experimentos descritos en el capítulo 4. Sin embargo una situación que se presenta es que el corpus Golem-UNIVERSUM proporciona información que aún no ha sido incluida en la solución que se propone en este trabajo. La información a la que se hace referencia es el contexto, o como se le denominó en el capítulo 2 la intención de la elocución. (Ver subsección 2.2.2.)

Con el propósito de determinar cuál es el efecto de agregar información del contexto a los modelos en Lógica de Markov con los que se ha trabajado, se plantean los experimentos descritos en este capítulo. El conjunto de experimentos es el mismo que se presentó en el capítulo anterior, sólo que los modelos consideran

información del contexto.

El contenido de este capítulo se encuentra estructurado como se describe a continuación. En la sección 5.1 se describen los experimentos que se realizan empleando las transcripciones manuales del corpus Golem-UNIVERSUM, así como los resultados obtenidos. En la sección 5.2 se explican los experimentos realizados y resultados obtenidos con las transcripciones automáticas de las elocuciones. En la sección 5.3 se presenta una discusión de la importancia de los resultados obtenidos.

## **5.1. Experimentos con transcripciones manuales**

Los experimentos descritos en esta sección son realizados empleando tanto la parte de desarrollo como de evaluación del corpus. Las transcripciones manuales de la parte de desarrollo del corpus son empleadas para el experimento de baseline y los experimentos parametrizadores. Para los experimentos de evaluación se emplea la parte del corpus con el mismo nombre, empleando tanto transcripciones manuales como automáticas.

### **5.1.1. Baseline**

Este experimento tiene como objetivo obtener una evaluación del desempeño del párser semántico implementado actualmente en el sistema Golem-UNIVERSUM mediante el empleo de la intención de la elocución además de la transcripción manual. Esta evaluación servirá como referencia para determinar, en primera instancia, el desempeño del párser del sistema que trabaja con la información del contexto, y después para comparar los resultados de los experimentos empleando Lógica de Markov.



Medida	Valor
Exactitud	81.50 %
F-score local	84.24 %
F-score global	84.58 %

Cuadro 5.1: Resultados de la evaluación del desempeño del pársers semántico del sistema Golem-UNIVERSUM con transcripciones manuales e información del contexto.

Las transcripciones manuales del corpus de desarrollo fueron sometidas al procesamiento semántico realizado por el pársers del sistema. Con la intención de obtener los valores de las métricas de evaluación, las representaciones asignadas por el pársers fueron comparadas con las representaciones asignadas manualmente a las transcripciones.

### Resultados

Con relación al valor de la exactitud mostrado en el cuadro 5.1 se puede afirmar que el pársers semántico del sistema Golem-UNIVERSUM presenta un buen desempeño durante el proceso de de asignar representaciones semánticas, pero se puede mejorar al emplear modelos en Lógica de Markov para el procesamiento semántico. El error se presenta en casi 2 de cada 10 representaciones asignadas a las transcripciones, lo que provoca que el sistema no sea capaz de dar una respuesta adecuada durante las interacciones. Si se compara este resultado con el obtenido en el capítulo 4 en el mismo experimento, el experimento baseline con transcripciones manuales sin intenciones, se presenta un mejor desempeño del pársers.

#### 5.1.2. Experimentos parametrizadores

Para los experimentos descritos en esta sección se emplearon 3 modelos: el base, el modelo bigramas y el modelo etiquetado. Estos modelos fueron modificados para

incluir la información del contexto. Esta nueva extensión del modelo se obtiene al agregar información nueva en forma de variables observables. Para poder agregar estas variables se definió el objeto intención en el dominio. Algunos ejemplos de este objeto son *edad*, *ok* y *característica*. Como se explicó en la sección 3.1, una transcripción puede estar relacionada a más de una intención. Para poder manejar esta condición se define el predicado *contexto/2*. Para el caso de que la transcripción tenga asociadas las expectativas *carta* y *característica* los predicados son:

$$\text{contexto}(1, \text{característica})$$

$$\text{contexto}(2, \text{carta})$$

Con el predicado *contexto/2* y el predicado *representacion/2* se puede definir la fórmula de primer orden

$$\phi_5 : \text{contexto}(q_1, i_1) \wedge \text{representacion}(\text{característica}, \text{valor})$$

define la relación entre la palabra  $i_1$  en la posición  $q_1$  con la representación semántica denotada por el predicado *representacion(característica, valor)*. El modelo base con información del contexto queda definido por  $\phi_1$  y  $\phi_5$ , donde:

$$\phi_1 : \text{palabra}(p_1, o_1) \wedge \text{representación}(\text{característica}, \text{valor})$$

Para la transcripción de la elocución de entrada *se trata de un planeta* y la expectativa (*carta, característica*) se puede obtener el grafo de factores que se muestra en la Figura 5.1.

El código empleado para definir el modelo base, así como el bigramas y el etiquetado, todos con información del contexto incluida, se encuentra en el apéndice B de este documento.

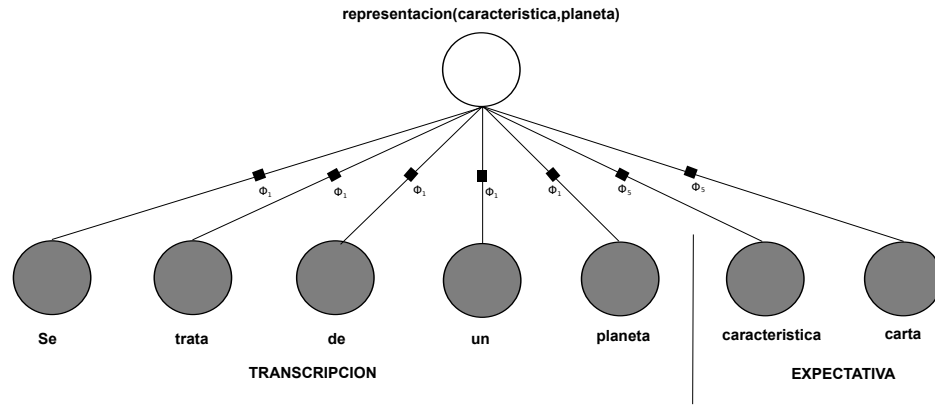


Figura 5.1: Modelo base extendido mediante la adición de información del contexto.

Para cada modelo se realiza una serie de entrenamientos, durante los cuales se varía el número de épocas en un rango de 1 a 20. Para cada entrenamiento se emplea en su totalidad las transcripciones manuales con sus intenciones de la parte de entrenamiento del corpus Golem-UNIVERSUM. El corpus se trabajó utilizando un proceso de validación cruzada como el que se muestra en la Figura 4.1.

### Resultados

Las curvas de aprendizaje de los modelos empleados para estos experimentos se muestran en la Figuras siguientes: 5.2 para el modelo base, 5.3 para el modelo bigramas y 5.4 para el modelo etiquetado.

En la curva de aprendizaje del modelo base se puede notar que el modelo supera por un 7 por ciento al baseline. Se observa también que la curva de aprendizaje para la exactitud mantiene un aumento en el aprendizaje dentro del rango de épocas de 1 a 9 para el modelo base. De este último valor hasta el 20, el aprendizaje presenta un comportamiento constante. En la gráfica de la Figura 5.2 también se puede observar que el punto de mejor aprendizaje del modelo base se presenta en la época 9.

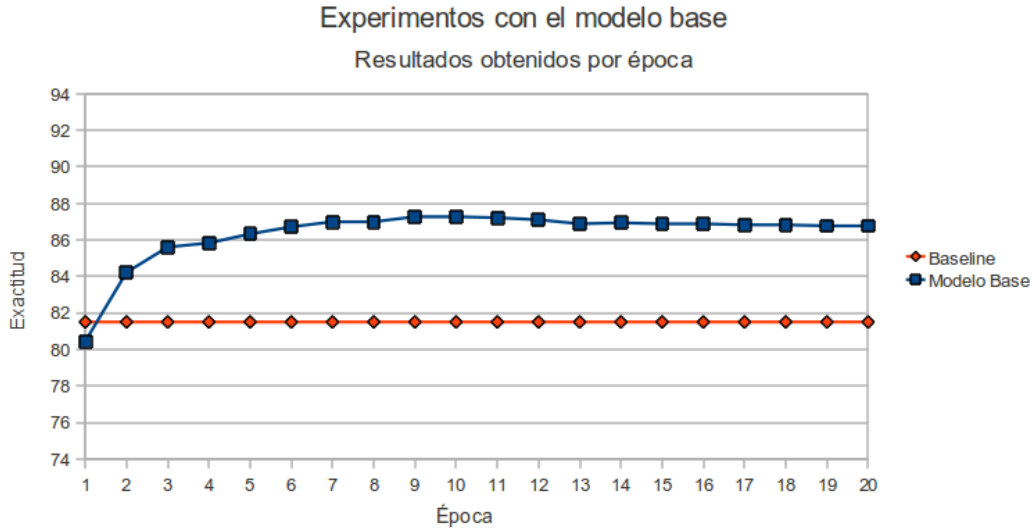


Figura 5.2: Curva de aprendizaje del modelo base empleando transcripciones manuales y expectativas.

En la curva de aprendizaje del modelo bigramas se puede ver que el modelo se queda a un 4 por ciento de superar al modelo base con respecto al punto donde presenta su mayor aprendizaje. Pero también en este punto se puede observar que el modelo bigramas supera por un 3 por ciento al baseline. El punto donde el modelo bigramas presenta su mayor aprendizaje es en la época 7. Se observa también que la curva de aprendizaje para la exactitud mantiene un aumento dentro del rango de 1 a 7 épocas para el modelo bigramas. La curva de aprendizaje presenta un comportamiento constante desde la época 8 hasta la época 20 como se observa en la Figura 5.3.

En la curva de aprendizaje del modelo etiquetado se puede observar que el modelo supera por un 4 por ciento al baseline en la época 10 donde presenta su mejor desempeño. Así también, se observa que no logra superar el desempeño

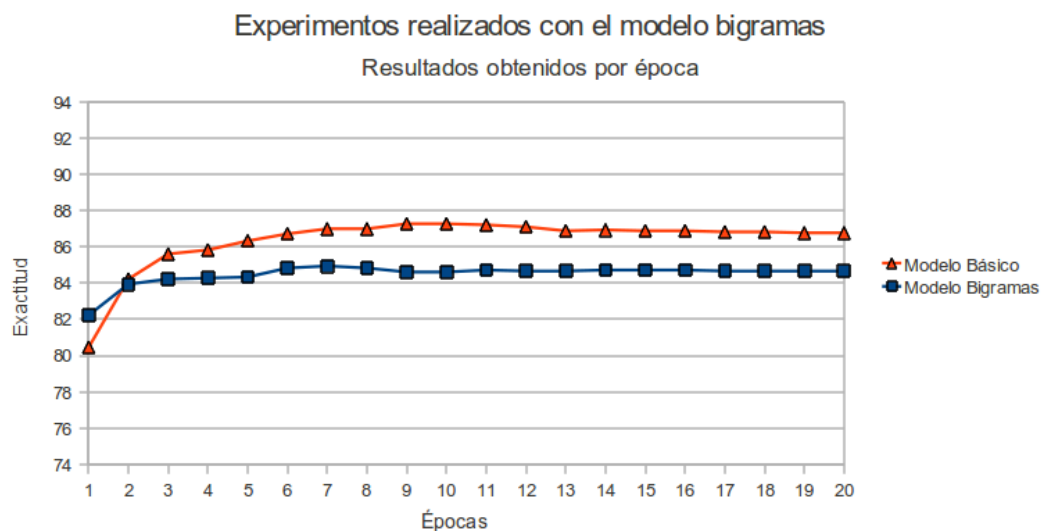


Figura 5.3: Curva de aprendizaje del modelo bigramas empleando transcripciones manuales y expectativas.

del modelo base. La curva de aprendizaje para la exactitud del modelo etiquetado, mostrada en la Figura 5.4, mantiene un aumento en el aprendizaje dentro del rango de épocas de 1 a 6, después de este valor presenta un comportamiento uniforme con pocas variaciones.

Se puede apuntar que los desempeños obtenidos presentan un comportamiento similar a los experimentos descritos en el capítulo 4. Es decir, para el caso del modelo base presenta un mejor desempeño comparado con el baseline. Lo mismo ocurre con los modelos bigramas y etiquetado con respecto al baseline, sin embargo no son capaces de superar el desempeño del modelo base. Esto se puede observar en el cuadro 5.2. Acorde a los datos del mismo cuadro, se puede notar que el modelo bigramas alcanza su máximo desempeño en menos épocas que el modelo base. Caso contrario con el del modelo etiquetado ya que tarda más en alcanzar su mejor desempeño.

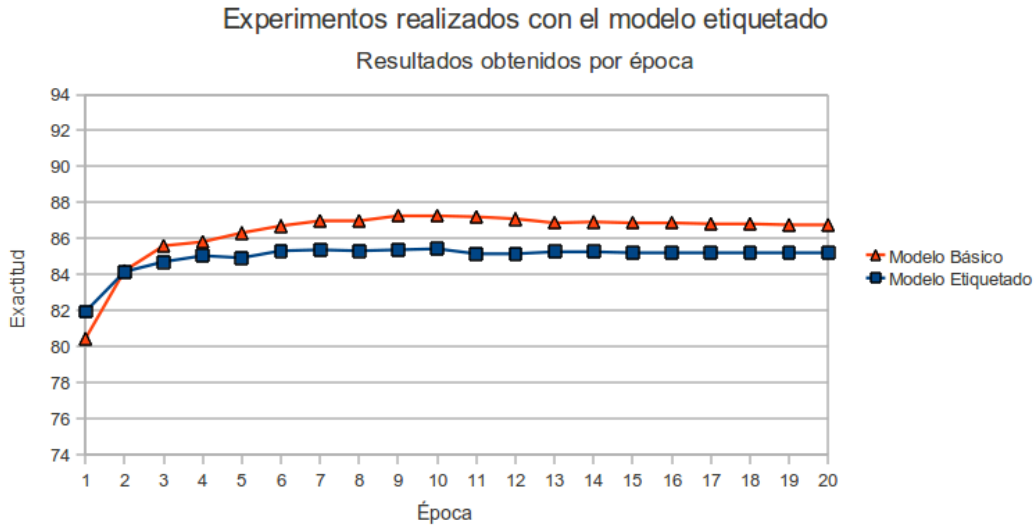


Figura 5.4: Curva de aprendizaje del modelo etiquetado empleando transcripciones manuales y expectativas.

Modelo	Modelo Básico	Modelo Bigramas	Modelo Etiquetado
Época	9	7	10
Exactitud	89.26 %	84.93 %	85.43 %
F-score local	92.11 %	90.80 %	90.86 %
F-score global	90.48 %	88.77 %	88.84 %

Cuadro 5.2: Comparación del desempeño de los modelos en las épocas donde presentan un mejor aprendizaje con transcripciones automáticas con expectativas.

### 5.1.3. Evaluación

Para estos experimentos el aprendizaje del modelo se realiza empleando la totalidad del corpus de entrenamiento. El parámetro de las épocas de entrenamiento se toma del cuadro 5.2 para cada modelo. Los resultados que se presentan se obtienen al evaluar los modelos con la parte de evaluación del corpus Golem-UNVERSUM.

Los modelos se evalúan en dos ocasiones. En la primera de ellas se emplean las transcripciones manuales con intenciones, esto con el propósito de mantener una

Modelo	Baseline	Modelo Base	Modelo Bigramas	Modelo Etiquetado
Época	-	9	7	10
exactitud	85.46 %	86.24 %	84.49 %	85.85 %
f-score local	89.24 %	91.57 %	90.54 %	91.24 %
f-score global	88.22 %	89.69 %	88.20 %	89.26 %

Cuadro 5.3: Comparación de las evaluaciones con transcripciones manuales de los modelos en las épocas donde presentan un mejor aprendizaje con transcripciones manuales y expectativas.

consistencia entre los datos de entrenamiento y evaluación. La segunda evaluación se lleva a cabo utilizando transcripciones automáticas porque el pársers semántico funciona con este tipo de transcripciones durante las interacciones. Los resultados de las evaluaciones con transcripciones manuales y automáticas se presentan en las Figuras 5.5 y 5.6 respectivamente. Los valores del baseline mostradas en las gráficas de evaluación, tanto de transcripciones manuales y automáticas, se obtuvieron al someter todas las transcripciones a un procesamiento semántico del pársers semántico del sistema Golem-UNIVERUSM.

En la evaluación con transcripciones manuales y expectativas (Figura 5.5) se presenta un mejor desempeño por parte de los modelos base y etiquetado respecto al baseline. En el caso del modelo bigramas se presenta un aprendizaje menor a 1 por ciento con respecto al baseline. La razón por la que esto ocurre es que los bigramas presentan una estructura muy compleja para poder trabajar con transcripciones manuales y sus elocuciones. Los valores de las métricas para la evaluación con transcripciones manuales se presentan en el cuadro 5.3.

En la evaluación con transcripciones automáticas y expectativas (Figura 5.6) se presenta un mejor desempeño por parte del modelo base respecto al baseline. En el caso de los modelo bigramas y etiquetado se presenta un aprendizaje menor con

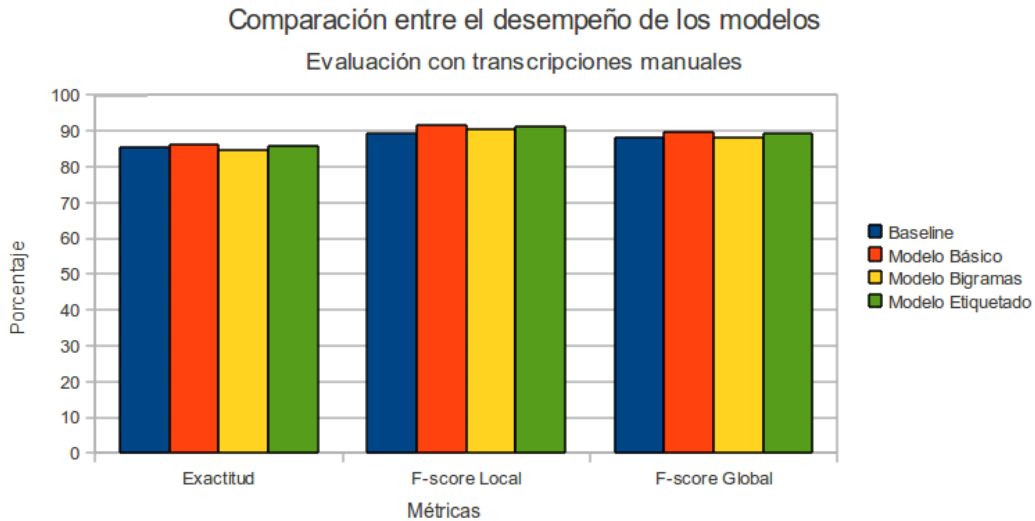


Figura 5.5: Resultados de la evaluación de los modelos empleando transcripciones manuales.

respecto al mismo dato. La razón por la que esto se presenta es, principalmente, la inconsistencia de los datos de entrenamiento y evaluación, ya que en algunos casos las relaciones presentes en la parte de evaluación del corpus no están incluidas en la parte de desarrollo del corpus. Los valores de las métricas obtenidos para esta evaluación se presentan en el cuadro 5.4

Modelo	Baseline	Modelo Base	Modelo Bigramas	Modelo Etiquetado
Época	-	9	7	10
exactitud	52.13 %	56.78 %	45.54 %	45.54 %
f-score local	58.72 %	65.11 %	54.45 %	55.62 %
f-score global	60.37 %	61.63 %	51.84 %	56.03 %

Cuadro 5.4: Comparación de las evaluaciones con transcripciones automáticas de los modelos en las épocas donde presentan un mejor aprendizaje con transcripciones manuales y expectativas.



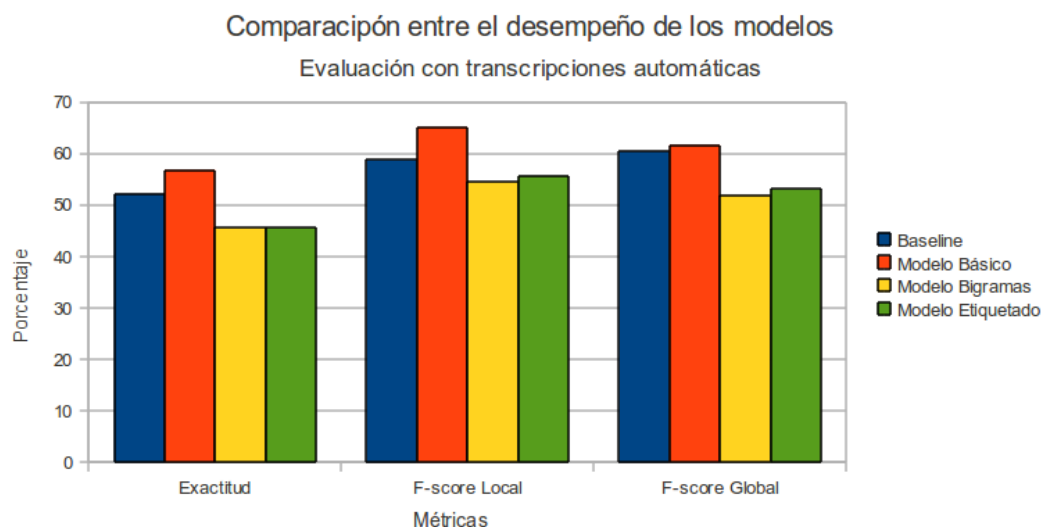


Figura 5.6: Resultados de la evaluación de los modelos empleando transcripciones automáticas.

## 5.2. Experimentos con transcripciones automáticas

Realizar experimentos con transcripciones automáticas y expectativas es de suma importancia principalmente por dos razones. Primero, las transcripciones automáticas del corpus son las más parecidas a las que se presentan durante las interacciones del sistema con los usuarios. Segundo, el párser del sistema Golem-UNIVERSUM incluye la información de las expectativas para realizar el procesamiento semántico.

Para realizar los experimentos que se describen en esta sección se emplean las transcripciones automáticas y expectativas, tanto la parte de entrenamiento como de evaluación del corpus. La parte de entrenamiento se emplea para el experimento de baseline y los experimentos parametrizadores. En el caso de los experimentos de evaluación se emplea la parte del corpus con el mismo nombre.

Medida	Valor
Exactitud	41.33 %
F-score local	46.87 %
F-score global	48.71 %

Cuadro 5.5: Resultados de la evaluación del desempeño del pársers semántico del sistema Golem-UNIVERSUM con transcripciones automáticas y expectativas.

### 5.2.1. Baseline

Los valores del desempeño del procesamiento semántico en el sistema Golem-UNIVERSUM cuando se emplean transcripciones automáticas con intenciones son de suma importancia para este trabajo, porque el pársers del sistema trabaja con transcripciones automáticas e información del contexto. Las mediciones de los resultados de este experimento, que se muestran en el cuadro 5.5, se obtuvieron al comparar las representaciones semánticas asignadas por el pársers semántico del sistema con las representaciones asignadas manualmente a las transcripciones.

Como se puede observar el desempeño obtenido es aún más deficiente que el obtenido en el mismo experimento con transcripciones automáticas e intenciones, a pesar de ello, sí presenta un mejor desempeño respecto al mismo experimento del capítulo 4, donde se emplean transcripciones manuales sin expectativas.

### 5.2.2. Experimentos parametrizadores

Estos experimentos consisten en una serie de entrenamientos con los tres modelos seleccionados (base, bigramas y etiquetados) con una variación en el parámetro de las épocas en un rango de 1 a 20. La finalidad con la que estos experimentos son realizados es conocer cuál es el desempeño de estos modelos con transcripciones automáticas e intenciones incluidas, además de identificar la época en la que pre-

sentan su mejor desempeño. Para llevar a cabo estos experimentos se emplea en su totalidad las transcripciones automáticas de la parte de entrenamiento del corpus Golem-UNIVERSUM, el cual se somete a un proceso de validación cruzada como el que se muestra en la Figura 4.1.

### **Resultados**

En las Figuras 5.7, 5.8 y 5.9 se muestran las curvas de aprendizaje para cada uno de los modelos empleados para estos experimentos, mientras que los valores del mejor desempeño se muestran en el cuadro 5.6.

El modelo base proporciona un mejor desempeño en un 16 por ciento basado en el baseline obtenido, lo que indica que el sistema pasa de 4 representaciones asignadas correctamente de cada 10 a casi 6 representaciones correctas por cada decena. Acorde a la curva de aprendizaje mostrada en la Figura 5.7, el modelo presenta un aprendizaje desde la época 1 hasta la 9. En el resto de las épocas se comporta de forma casi constante con muy pocas variaciones. El mejor desempeño de este modelo se presenta en la época 13.

El modelo bigramas presenta un mejor desempeño en un 16 por ciento respecto al baseline. Para el caso del modelo base, el modelo bigramas no logra superar su desempeño. La curva de aprendizaje mostrada en la Figura 5.8 indica que el modelo presenta un aprendizaje desde la época 1 hasta la 12, pero con un leve descenso en los valores entre las épocas 5 y 9. De la época 12 a la época 20 el aprendizaje del modelos se comporta de forma casi constante, ya que presenta algunas variaciones. El mejor desempeño del modelo bigramas se presenta en la época 17.

El modelo etiquetado, al igual que el modelo bigramas, presenta un mejor desempeño respecto al baseline en un 14 por ciento. Por otro lado el modelo etiquetado no supera el desempeño del modelo base. Como se aprecia en la Figura

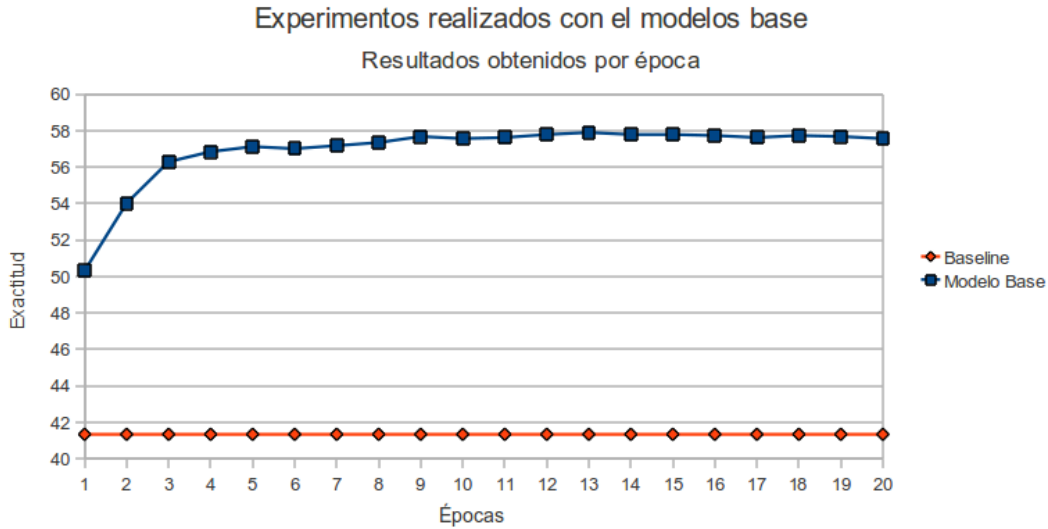


Figura 5.7: Curva de aprendizaje del modelo base empleando transcripciones automáticas y expectativas.

5.9, el aprendizaje del modelo etiquetado se presenta de forma irregular, ya que se presentan descensos en los valores de la exactitud en el rango de la época 6 a la 9 y de la época 11 a la 15. El mejor desempeño de este modelo se presenta en la época 19.

Los tres modelos presentan un desempeño superior con respecto al baseline. Sin embargo el modelo bigramas y el modelo etiquetado no mejoran en comparación con el modelo base. Un punto a favor del modelo base es que requiere de un menor número de épocas para alcanzar su máximo aprendizaje, lo que se traduce en un menor tiempo de entrenamiento.

### 5.2.3. Evaluación

Para la evaluación que se desarrolla en esta sección, el entrenamiento de los modelos utilizados (base, bigramas y etiquetado) se realiza empleando la totali-

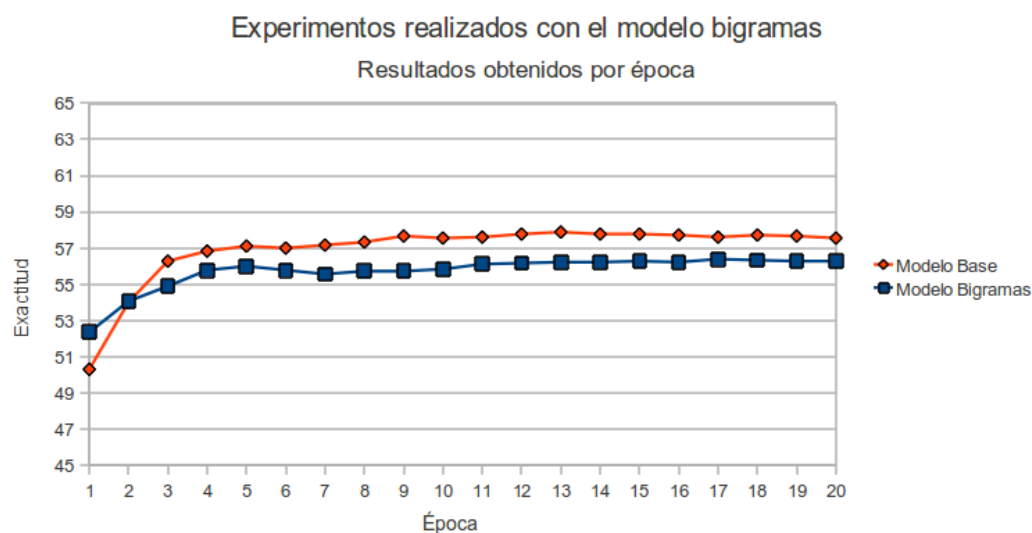


Figura 5.8: Curva de aprendizaje del modelo bigramas empleando transcripciones automáticas y expectativas.

Modelo	Modelo Básico	Modelo Bigramas	Modelo Etiquetado
Época	13	17	19
Exactitud	57.88 %	56.39 %	55.72 %
F-score local	69.72 %	66.95 %	66.76 %
F-score global	66.30 %	62.39 %	62.28 %

Cuadro 5.6: Comparación del desempeño de los modelos en las épocas donde presentan un mejor aprendizaje con transcripciones automáticas y expectativas.

dad las transcripciones automáticas de la parte de desarrollo del corpus. Para el entrenamiento de cada modelo se toman en cuenta las épocas que se muestran en el cuadro 5.6. Mientras que los resultados presentados se obtienen al evaluar estos modelos con las transcripciones automáticas de la parte de evaluación del corpus Golem-UNIVERSUM.

Los resultados arrojados por esta evaluación presentan la misma tendencia de los demás resultados obtenidos en este trabajo. Es decir, el desempeño de los tres

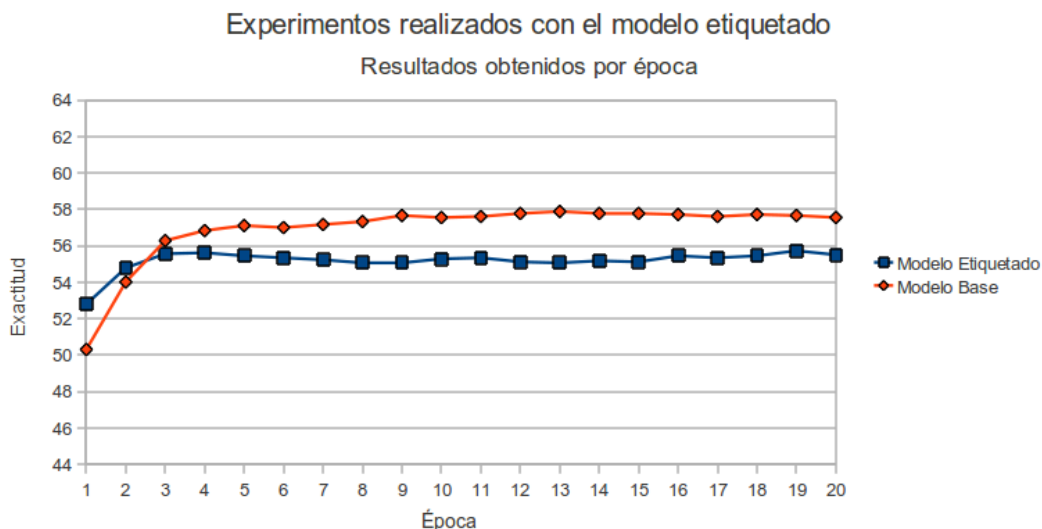


Figura 5.9: Curva de aprendizaje del modelo etiquetado empleando transcripciones automáticas y expectativas.

modelos propuestos superan el desempeño del baseline, pero los modelos bigramas y etiquetado no son capaces de superar los valores obtenidos con el modelo base.

Los resultados de esta evaluación se presentan en la gráfica de la Figura 5.10. El baseline que se muestra en esta gráfica se obtuvo al someter a un procesamiento semántico todas las transcripciones automáticas con intenciones de la parte de evaluación del corpus. Como se puede apreciar, si se comparan las gráficas de las Figuras 4.12 y 5.10, la información del contexto agregada a los modelos se ve reflejada en un aumento en el desempeño de los modelos al momento de llevar a cabo el procesamiento semántico.

Los resultados de esta evaluación son de suma importancia para este trabajo ya que nos indica que se puede mejorar hasta en un 10 por ciento el desempeño del parser semántico del sistema Golem-UNIVERSUM, como se puede comprobar en

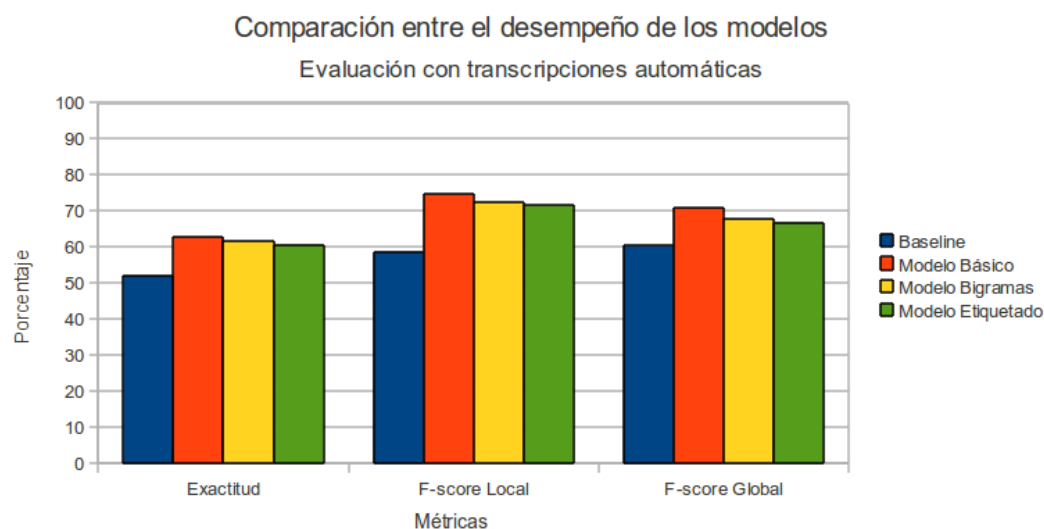


Figura 5.10: Resultados de la evaluación de los modelos empleando transcripciones automáticas y expectativas.

el cuadro 5.7. Esto considerando que el párser implementado con Lógica de Markov empleara la misma información utilizada por el párser actualmente instalado, es decir las transcripciones automáticas y la información del contexto.

Modelo	Baseline	Modelo Base	Modelo Bigramas	Modelo Etiquetado
Época	-	13	17	19
Exactitud	52.13 %	62.79 %	61.62 %	60.65 %
F-score local	58.72 %	74.70 %	72.48 %	71.60 %
F-score global	60.37 %	70.86 %	67.76 %	66.83 %

Cuadro 5.7: Comparación de las evaluaciones con transcripciones automáticas de los modelos en las épocas donde presentan un mejor aprendizaje con transcripciones automáticas y expectativas.

### 5.3. Discusión

En este capítulo se mostraron los experimentos planteados con la finalidad de determinar el modelo en Lógica de Markov que llevará a cabo de la mejor manera el procesamiento semántico empleando información del contexto. Esta información se encuentra en el corpus representada por las intenciones relacionadas a cada una de las transcripciones que se tienen. La importancia de estos experimentos radica en dos razones. La primera de las razones es que los experimentos demuestran que la información del contexto puede ser de utilidad para llevar a cabo el procesamiento semántico. La segunda razón es que para la implementación del pársers basado en Lógica de Markov no se requiere cambiar el diseño que el sistema Golem-UNIVERSUM tiene actualmente.

Los modelos presentaron un mayor desempeño en comparación con el pársers implementado actualmente en el sistema Golem-UNIVERSUM, considerando para ambos casos las intenciones de las transcripciones. El modelo base presenta el mejor desempeño superando a los modelos bigramas y etiquetado. Esta tendencia se presenta en todas las evaluaciones realizadas, excepto en el caso donde emplean transcripciones automáticas para la evaluación de modelos entrenados con transcripciones manuales. La razón de que esto ocurra es la inconsistencia entre los datos empleados para el entrenamiento y evaluación. Otro caso donde se presenta este cambio en la tendencia es en el caso del modelo bigramas cuando es entrenado y evaluado con transcripciones manuales.

En base a los resultados de las evaluaciones realizadas, tanto en este capítulo como en el capítulo 4, se puede concluir que la información del contexto ayuda a un mejor entrenamiento de los modelos. Esto no solo se presenta al comparar los mo-



delos en Lógica de Markov, sino también se puede notar en el caso de los diferentes valores baseline obtenidos durante la realización de todos los experimentos.

Como conclusión del trabajo presentado en este capítulo se tiene que la implementación de cualquiera de los modelos se puede traducir en un mejor desempeño del pársers semántico, lo que a su vez produce un mejor desempeño del sistema en su totalidad. En el caso del modelo base se debe considerar que el entrenamiento del modelo requiere de menos tiempo que en el caso de los modelos bigramas y etiquetado.

Con la información obtenida de los experimentos mostrados, tanto en el capítulo 4 como en el presente capítulo, se pueden obtener algunas conclusiones relacionadas con los objetivos que se plantearon en este trabajo. En el siguiente capítulo se indican cuales son las conclusiones obtenidas, las aportaciones de este trabajo así como las líneas de trabajo a futuro que se pueden derivar de esta tesis.



## Capítulo 6

# Conclusiones

La presencia de los sistemas de diálogo en la vida cotidiana lentamente ha mostrado un crecimiento en los últimos años. Es por esta razón que el desarrollo de sistemas suficientemente robustos para mantener una conversación coherente con un ser humano se ha convertido en un objetivo de la inteligencia artificial. Este trabajo se propuso con la finalidad de aumentar la capacidad de respuesta del sistema de diálogo implementando en el módulo Golem-UNIVERSUM mediante la mejora del pársers semántico con el que cuenta actualmente.

Con el fin de detallar las conclusiones y las aportaciones de este trabajo el capítulo presenta la siguiente estructura: en la sección 6.1 se presentan las conclusiones del trabajo desarrollado en esta tesis; en la sección 6.2 se explican las contribuciones de este trabajo al sistema Golem-UNIVERSUM y en la sección 6.3 se presentan líneas de trabajo a futuro que se pueden derivar de esta tesis.

## 6.1. Conclusiones

Este trabajo se sustenta en la aplicación de modelos en Lógica de Markov para llevar a cabo el procesamiento semántico en el sistema Golem-UNIVERSUM. Lógica de Markov presenta características importantes que son de utilidad para este trabajo, por ejemplo el manejo de incertidumbre así como un buen desempeño en el manejo de información nueva o desconocida en el dominio.

Los resultados de la aplicación de estos modelos en el sistema Golem-UNIVERSUM son de utilidad para llegar a las siguientes conclusiones.

Los modelos en Lógica de Markov propuestos muestran que son capaces de modelar de una manera robusta las relaciones entre las transcripciones de las elocuciones de entrada al sistema y las representaciones semánticas adecuadas. Prueba de esto es que los tres modelos seleccionados durante el desarrollo de este trabajo muestran un buen desempeño al llevar a cabo el procesamiento semántico, superior al desempeño que presenta el pársers actual.

Se concluye también que el corpus Golem-UNIVERSUM es de utilidad para el entrenamiento de los modelos propuestos considerando que el tamaño del corpus es pequeño. En el caso del modelo base, la información que proporciona el corpus es suficiente para obtener un buen entrenamiento del modelo. Por otro lado, los modelos bigramas y etiquetados pueden alcanzar un mejor desempeño con una mayor cantidad de información para el entrenamiento, es decir, con una mayor cantidad de transcripciones en el corpus.

Las evaluaciones de los modelos, así como sus entrenamientos, se plantearon con la idea de hacerlos comparables y que las comparaciones realizadas fueran lo más justos posibles. Lo mismo ocurre con las métricas que se proponen para cuantificar

Medida	Valor
Exactitud	52.13 %
F-score local	58.17 %
F-score global	60.37 %

Cuadro 6.1: Valores del desempeño del párser semántico del sistema Golem-UNIVERSUM con transcripciones automáticas y expectativas.

los resultados.

La aplicación de modelos en Lógica de Markov para llevar a cabo procesamiento del semántico se había realizado con éxito relativo en el idioma inglés. Una de las aportaciones de este trabajo es la premisa que se puede aplicar también al idioma español para llevar a cabo la misma tarea. Los resultados indican que se puede aplicar con resultados significativos.

Las conclusiones anteriores están relacionadas directamente con los objetivos específicos planteados. En lo que respecta al objetivo general de este trabajo se puede considerar como alcanzado ya que el desempeño del párser semántico se vio incrementado como se explica en la siguiente sección.

## 6.2. Contribuciones

Como se indicó al inicio de este trabajo el sistema Golem-UNIVERSUM presentaba un desempeño deficiente durante el procesamiento semántico de las transcripciones de las elocuciones. Los valores que miden este desempeño se muestran en el cuadro 6.1. El valor de la exactitud mostrado en el cuadro indica que el párser semántico es capaz de asignar 5 representaciones correctas de cada 10 elocuciones de entrada. Lo que se traduce en una repetitiva respuesta incorrecta por parte del sistema, provocando que las interacciones se vuelvan tediosas.

	Baseline	Modelo Básico	Modelo Bigramas	Modelo Etiquetado
Época	-	13	17	19
Exactitud	52.13 %	62.79 %	61.62 %	60.65 %
F-score local	58.17 %	74.70 %	72.48 %	71.60 %
F-score global	60.37 %	70.86 %	67.76 %	66.83 %
Mejora	-	20.44 %	18.20 %	16.23 %

Cuadro 6.2: Comparación entre los valores obtenidos durante la evaluación de los modelos y el baseline, ambos con transcripciones automáticas y expectativas.

Los resultados obtenidos de los experimentos realizados en este trabajo indican que los modelos en Lógica de Markov son capaces de incrementar a 6 representaciones asignadas correctamente por cada 10 elocuciones de entrada. Los valores de las métricas de evaluación para cada uno de los modelos se muestra en el cuadro 6.2

En la última fila del cuadro 6.2 se indica cuál fue la mejora en el desempeño de cada modelo respecto al baseline que se muestra en el cuadro 6.1. Como se puede observar los valores obtenidos indican que el desempeño del pársers semántico aumenta con el uso de los modelos en Lógica de Markov. Esto se puede ver reflejado en interacciones mas fluidas y menor cantidad de respuestas erróneas por parte del sistema al usuario.

Para llegar a los resultados y conclusiones presentados fue necesario el desarrollo de los experimentos que se muestran en el cuadro 6.3. Los programas desarrollados para llevar a cabo los experimentos descritos en el cuadro anterior se encuentran disponibles en el sitio<sup>1</sup> que fue creado con esta finalidad.

<sup>1</sup><http://code.google.com/p/semantic-processing/>

	Sin Expectativas	Con Expectativas
Transcripciones Manuales	Baseline Desarrollo Evaluaciones	Baseline Desarrollo Evaluaciones
Transcripciones Automáticas	Baseline Desarrollo Evaluaciones	Baseline Desarrollo Evaluaciones

Cuadro 6.3: Experimentos realizados empleando los tipos de transcripciones disponibles así como las expectativas.

### 6.3. Trabajo a futuro

El trabajo desarrollado así como los resultados indican que existen algunas líneas que se pueden seguir para trabajo futuro. A continuación una descripción de las que se encuentran contextualizadas en el sistema Golem-UNIVERSUM.

- **Resistencia al dominio:** se puede trabajar sobre la adición de información al dominio, por ejemplo agregar una o más cartas al juego “Adivina la carta”. Sería necesario obtener el desempeño de los modelos en Lógica de Markov, modificarlos o si se considerara necesario plantear nuevos modelos.
- **Elocuciones de entrada más complejas:** Para este caso, es necesario proponer una representación semántica para poder representar la intención del usuario. En lo que respecta a los modelos en Lógica de Markov se tendrían que replantear para ser capaces de trabajar con la nueva representación propuesta.
- **El caso de la negación en las elocuciones de entrada:** Al igual que el caso anterior es necesario plantear una representación semántica con la que se pueda capturar la negación en una estructura formal. El replanteamiento

de los modelos en Lógica de Markov es inevitable.

Tomando en cuenta las conclusiones de este trabajo otra línea de trabajo es explorar el uso de modelos en Lógica de Markov en diferentes dominios dentro del idioma español.



## Apéndice A

# Resultados de los experimentos con modelos extendidos

Como se indica en la sección 4.1.3, se tienen tres modelos en Lógica de Markov: el modelo base, el modelo bigrama y el modelo etiquetado. Estos tres modelos fueron combinados de manera que se generaron los modelos que se listan a continuación:

- Base + Bigramas.
- Base + Etiquetado.
- Base + Bigramas + Etiquetado.

En los modelos donde se emplea el etiquetado se manejaron 4 diferentes posiciones de las palabras que son:

- Etiqueta de la palabra actual.
- Etiqueta de la palabra actual + Etiqueta de la palabra anterior.

- Etiqueta de la palabra actual + Etiqueta de la palabra siguiente.
- Etiqueta de la palabra actual + Etiqueta de la palabra anterior + Etiqueta de la palabra siguiente.

La combinación de los modelos listados nos da un total de 9 modelos extendidos. Cada modelo fue entrenado en un rango de épocas de 7 a 11. Los resultados obtenidos de estos entrenamientos se muestran en el cuadro A.1

Métricas	Épocas				
	7	8	9	10	11
Modelo Base + Bigramas					
Exactitud	84.50 %	84.55 %	84.61 %	84.61 %	84.55 %
F-score local	90.34 %	90.37 %	90.39 %	90.39 %	90.39 %
F-score global	88.28 %	88.31 %	88.35 %	88.35 %	88.33 %
Modelo Base + Etiquetado Posición Actual					
Exactitud	84.28 %	84.33 %	84.33 %	84.33 %	84.33 %
F-score local	90.23 %	90.23 %	90.20 %	90.20 %	90.20 %
F-score global	88.11 %	88.11 %	88.07 %	88.07 %	88.07 %
Modelo Base + Etiquetado Posición Anterior					
Exactitud	76.25 %	77.52 %	81.29 %	81.29 %	80.52 %
F-score local	81.06 %	82.39 %	86.68 %	86.66 %	85.87 %
F-score global	80.08 %	81.12 %	84.84 %	84.83 %	84.07 %
Modelo Base + Etiquetado Posición Siguiete					
Exactitud	84.39 %	84.50 %	84.28 %	84.28 %	84.28 %
F-score local	90.17 %	90.34 %	90.14 %	90.20 %	90.20 %
F-score global	87.93 %	88.14 %	87.91 %	87.99 %	87.99 %
Modelo Base + Etiquetado Posición Anterior y Siguiete					
Exactitud	84.16 %	84.16 %	84.16 %	84.11 %	84.11 %
F-score local	89.95 %	89.95 %	89.95 %	89.92 %	89.92 %
F-score global	87.74 %	87.74 %	87.74 %	87.71 %	87.71 %
Modelo Base + Bigramas + Etiquetado Posición Actual					
Exactitud	84.50 %	84.50 %	84.50 %	84.50 %	84.50 %
F-score local	90.45 %	90.45 %	90.42 %	90.42 %	90.42 %
F-score global	88.33 %	88.33 %	88.31 %	88.31 %	88.31 %
Modelo Base + Bigramas + Etiquetado Posición Anterior					
Exactitud	83.73 %	83.73 %	83.73 %	83.73 %	83.39 %
F-score local	89.03 %	89.00 %	89.00 %	89.12 %	88.89 %
F-score global	86.82 %	86.78 %	86.78 %	86.87 %	86.68 %
Modelo Base + Bigramas + Etiquetado Posición Siguiete					
Exactitud	84.66 %	84.66 %	84.66 %	84.66 %	84.66 %
F-score local	90.26 %	90.26 %	90.23 %	90.23 %	90.23 %
F-score global	88.10 %	88.10 %	88.07 %	88.07 %	88.07 %
Modelo Base + Bigramas + Etiquetado Posición Anterior y Siguiete					
Exactitud	84.11 %	84.11 %	84.05 %	84.05 %	84.05 %
F-score local	90.12 %	90.12 %	90.09 %	90.06 %	90.03 %
F-score global	87.98 %	87.96 %	87.92 %	87.89 %	87.87 %

Cuadro A.1: Resultados de los entrenamientos realizados con los modelos extendidos en Lógica de Markov.



## Apéndice B

# Modelos en Lógica de Markov

Para emplear el programa The Beast para el entrenamiento y la evaluación de los modelos, se requiere que estos se encuentren definidos en el lenguaje del programa. El código B.1 presenta la definición del modelo base con expectativas. Los códigos B.2 y B.3 presentan los códigos de definición del modelo bigramas y etiquetado, ambos con expectativas. Los modelos descritos fueron empleados para los experimentos del capítulo 5. Para los experimentos del capítulo 4 solo se requiere eliminar los predicados que hacen referencia a las intenciones, así como los pesos que incluyen estos predicados.

```
1 predicate word: Int x Word;
2 predicate goal: Feat x Val;
3 predicate inten: Int x Inten;
4
5 factor [0]: for Word w if word(_,w): |Feat f, Val v: goal(f,v)| <= 1;
6
7 weight w_feat: Feat x Word -> Double;
8 factor: for Word w, Feat f, Val v
```

```

9   if word(_,w) add [goal(f,v)] * w_feat(f,w);
10
11  weight w_val: Val x Word -> Double;
12  factor: for Word w, Feat f, Val v
13    if word(_,w) add [goal(f,v)] * w_val(v,w);
14
15  weight w_goal: Word x Feat x Val -> Double;
16  factor: for Word w, Feat f, Val v
17    if word(_,w) add [goal(f,v)] * w_goal(w,f,v);
18
19  weight w_inten: Inten x Feat x Val -> Double;
20  factor: for Inten i, Feat f, Val v
21    if inten(_,i) add [goal(f,v)] * w_inten(i,f,v);
22
23  observed: word,inten;
24  hidden: goal;

```

Código B.1: Definición del modelo base con expectativas

```

1  predicate word: Int x Word;
2  predicate pos: Int x Label;
3  predicate goal: Feat x Val;
4  predicate inten: Int x Inten;
5
6  factor[0]: for Word w if word(_,w): |Feat f, Val v: goal(f,v)| <= 1;
7
8  weight w_feat: Feat x Word -> Double;
9  factor: for Word w, Feat f, Val v
10   if word(_,w) add [goal(f,v)] * w_feat(f,w);
11
12  weight w_val: Val x Word -> Double;

```

```

13 factor: for Word w, Feat f, Val v
14   if word(_,w) add [goal(f,v)] * w_val(v,w);
15
16 weight w_goal: Word x Feat x Val -> Double;
17 factor: for Word w, Feat f, Val v
18   if word(_,w) add [goal(f,v)] * w_goal(w,f,v);
19
20 weight w_feat_big: Feat x Word x Word -> Double;
21 factor: for Int p, Word w1, Word w2, Feat f, Val v
22   if word(p,w1) & word(p+1,w2) add [goal(f,v)]*w_feat_big(f,w1,w2);
23
24 weight w_val_big: Val x Word x Word -> Double;
25 factor: for Int p, Word w1, Word w2, Feat f, Val v
26   if word(p,w1) & word(p+1,w2) add [goal(f,v)]*w_val_big(v,w1,w2);
27
28 weight w_goal_big: Word x Word x Feat x Val -> Double;
29 factor: for Int p, Word w1, Word w2, Feat f, Val v
30   if word(p,w1)&word(p+1,w2) add [goal(f,v)]*w_goal_big(w1,w2,f,v);
31
32 weight w_inten: Inten x Feat x Val -> Double;
33 factor: for Inten i, Feat f, Val v
34   if inten(_,i) add [goal(f,v)] * w_inten(i,f,v);
35
36 observed: word, pos, inten;
37 hidden: goal;

```

Código B.2: Definición del modelo bigramas con expectativas

```

1 predicate word: Int x Word;
2 predicate pos: Int x Label;
3 predicate goal: Feat x Val;

```

```

4 predicate inten: Int x Inten;
5
6 factor [0]: for Word w if word(_,w): |Feat f, Val v: goal(f,v)| <= 1;
7
8 weight w_feat: Feat x Word -> Double;
9 factor: for Word w, Feat f, Val v
10   if word(_,w) add [goal(f,v)] * w_feat(f,w);
11 weight w_val: Val x Word -> Double;
12 factor: for Word w, Feat f, Val v
13   if word(_,w) add [goal(f,v)] * w_val(v,w);
14
15 weight w_goal: Word x Feat x Val -> Double;
16 factor: for Word w, Feat f, Val v
17   if word(_,w) add [goal(f,v)] * w_goal(w,f,v);
18
19 weight w_feat_big: Feat x Word x Word -> Double;
20 factor: for Int p, Word w1, Word w2, Feat f, Val v
21   if word(p,w1) & word(p+1,w2) add [goal(f,v)] * w_feat_big(f,w1,w2);
22
23 weight w_val_big: Val x Word x Word -> Double;
24 factor: for Int p, Word w1, Word w2, Feat f, Val v
25   if word(p,w1) & word(p+1,w2) add [goal(f,v)] * w_val_big(v,w1,w2);
26
27 weight w_goal_big: Word x Word x Feat x Val -> Double;
28 factor: for Int p, Word w1, Word w2, Feat f, Val v
29   if word(p,w1)&word(p+1,w2) add [goal(f,v)] * w_goal_big(w1,w2,f,v);
30
31 weight w_feat_postag: Feat x Word x Label -> Double;
32 factor: for Int p, Word w, Label l, Feat f, Val v
33   if word(p,w) & pos(p,l) add [goal(f,v)] * w_feat_postag(f,w,l);

```



```

34
35 weight w_val_postag: Val x Word x Label -> Double;
36 factor: for Int p, Word w, Label l, Feat f, Val v
37   if word(p,w) & pos(p,l) add [goal(f,v)]*w_val_postag(v,w,l);
38
39 weight w_goal_postag: Word x Label x Feat x Val -> Double;
40 factor: for Int p, Word w, Label l, Feat f, Val v
41   if word(p,w) & pos(p,l) add [goal(f,v)]*w_goal_postag(w,l,f,v);
42
43 weight w_feat_pos_dif: Feat x Word x Label -> Double;
44 factor: for Int p, Word w, Label l1, Feat f, Val v
45   if word(p,w)&pos(p+1,l1) add [goal(f,v)]*w_feat_pos_dif(f,w,l1);
46
47 weight w_val_pos_dif: Val x Word x Label -> Double;
48 factor: for Int p, Word w, Label l1, Feat f, Val v
49   if word(p,w)&pos(p+1,l1) add [goal(f,v)]*w_val_pos_dif(v,w,l1);
50
51 weight w_goal_pos_dif: Feat x Val x Word x Label -> Double;
52 factor: for Int p, Word w, Label l1, Feat f, Val v
53   if word(p,w)&pos(p+1,l1) add [goal(f,v)]*w_goal_pos_dif(f,v,w,l1);
54
55 weight w_inten: Inten x Feat x Val -> Double;
56 factor: for Inten i, Feat f, Val v
57   if inten(_,i) add [goal(f,v)] * w_inten(i,f,v);
58
59 observed: word, pos, inten;
60 hidden: goal;

```

Código B.3: Definición del modelo etiquetado con expectativas



# Bibliografía

- Allen, J., Ferguson, G., Miller, B. W., Ringger, E. K., and Sikorski-Zollo, T. (2000). Dialogue systems: From theory to practice in TRAINS-96. In *Handbook of Natural Language Processing*, pages 347–376. New York.
- Bay, H., Ess, A., Tuytelaars, T., and Gool, L. J. V. (2008). Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding*, 110(3):346–359.
- Bradski, G. and Kaehler, A. (2008). *Learning OpenCV: Computer Vision with the OpenCV Library*. O’Reilly, Cambridge, MA.
- Cheyner, A. and Martin, D. L. (2001). The Open Agent Architecture. *Autonomous Agents and Multi-Agent Systems*, 4(1/2):143–148.
- Cohn, T. (2007). *Scaling Conditional Random Fields for Natural Language Processing*. PhD thesis, Faculty of Engineering, University of Melbourne.
- Ferguson, G. and Allen, J. F. (1998). TRIPS: An Integrated Intelligent Problem-Solving Assistant. In *AAAI/IAAI*, pages 567–572.
- Gustafson, J., Lindberg, N., and Lundeberg, M. (1999). The August Spoken Dialogue System. In *In Proceedings of Eurospeech’99*, pages 1151–1154.

- Huerta, J. M., Chen, S., and Stern, R. M. (1999). The 1998 Carnegie Mellon University SPHINX-3 Spanish Broadcast News Transcription System. In *In Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*.
- Jurafsky, D. and Martin, J. H. (2008). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*. Prentice Hall, second edition.
- Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning, ICML '01*, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.
- Manning, C. D. and Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA.
- McDonald, R., Crammer, K., and Pereira, F. (2005). Online large-margin training of dependency parsers. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, ACL '05*, pages 91–98, Stroudsburg, PA, USA. Association for Computational Linguistics.
- McTear, M. F. (2004). *Spoken Dialogue Technology: Toward the Conversational User Interface*. Springer Verlag.
- Meza, I. (2009). *Statistical Semantic Processing usign Markov Logic*. PhD thesis, School of Informatics, University of Edinburgh.

- Meza, I., Elia, P., Salinas, L., Aviles, H., and Pineda, L. A. (2010). A Multimodal Dialogue System for Playing the Game "Guess the card". *Procesamiento de Lenguaje Natural*, (44):131–138.
- Olson, D. L. and Delen, D. (2008). *Advanced Data Mining Techniques*. Springer Publishing Company, Incorporated, 1st edition.
- Pineda, L., Castellanos, H., Cuétara, J., Galescu, L., Juárez, J., Llisterri, J., Pérez, P., and Villaseñor, L. (2009). The Corpus DIMEx100: transcription and evaluation. *Language Resources and Evaluation*.
- Pineda, L., Meza, I., and Salinas, L. (2010). Dialogue Model Specification and Interpretation for Intelligent Multimodal HCI. In *Proceedings IBERAMIA 2010. LNCS (LNAI)*, pages 20–29.
- Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286.
- Ratnaparkhi, A. (1999). Learning to Parse Natural Language with Maximum Entropy Models. *Mach. Learn.*, 34:151–175.
- Richardson, M. and Domingos, P. (2006). Markov logic networks. *Machine Learning*, 62(1-2):107–136.
- Riedel, S. (2008). Improving the Accuracy and Efficiency of MAP Inference for Markov Logic. In McAllester, D. A. and Myllymäki, P., editors, *UAI*, pages 468–475. AUAI Press.
- Seneff, S., Lau, R., Pao, C., and Zue, V. (2003). Galaxy-II: A Reference Architecture.

- Woods, W. A. (1973). Progress in natural language understanding: an application to lunar geology. In *AFIPS '73: Proceedings of the June 4-8, 1973, national computer conference and exposition*, pages 441–450, New York, NY, USA. ACM.
- Zettlemoyer, L. S. and Collins, M. (2007). Online learning of relaxed CCG grammars for parsing to logical form. In *In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL-2007)*, pages 678–687.