
Creación y evaluación de un corpus de
estructuras narrativas mediante
colaboración masiva entre usuarios de un
minijuego web



Proyecto Fin de Máster en Sistemas Inteligentes

Nahum Álvarez Ayerza

Director: Pablo Gervás Gómez-Navarro

Colaborador: Federico Peinado Gil

Máster en Investigación en Informática

Facultad de Informática

Universidad Complutense de Madrid

Curso 2009-2010

Documento maquetado con T_EX!S v.1.0.

Creación y evaluación de un corpus de estructuras narrativas mediante colaboración masiva entre usuarios de un minijuego web

*Proyecto Fin de Máster en Sistemas Inteligentes
que presenta para optar al título de
Máster en Investigación en Informática*
Nahum Álvarez Ayerza

Trabajo dirigido por
Pablo Gervás Gómez-Navarro
siendo colaborador externo de dirección
Federico Peinado Gil

**Máster en Investigación en Informática
Facultad de Informática
Universidad Complutense de Madrid**

Curso 2009-2010
Madrid, 13 de septiembre de 2010

Autorización

El abajo firmante, matriculado en el Máster en Investigación en Informática de la Facultad de Informática, autoriza a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor el presente Trabajo Fin de Máster: «Creación y evaluación de un corpus de estructuras narrativas mediante colaboración masiva entre usuarios de un minijuego web», realizado durante el curso académico 2009-2010 bajo la dirección de Pablo Gervás Gómez-Navarro y con la colaboración externa de dirección de Federico Peinado Gil en el Departamento de Ingeniería del Software e Inteligencia Artificial, y a la Biblioteca de la UCM a depositarlo en el Archivo Institucional E-Prints Complutense con el objeto de incrementar la difusión, uso e impacto del trabajo en Internet y garantizar su preservación y acceso a largo plazo.

Fdo. Nahum Álvarez Ayerza

*A los chicos de Echidna,
y a Andrés, Edu, Miguel y Tole*

Agradecimientos

*El éxito es aprender a ir de fracaso en
fracaso sin desesperarse.*

Winston Churchill

Quiero agradecer el apoyo brindado durante la realización de este trabajo a mi familia, por aguantarme mientras me pasaba los meses encerrado por las tardes en la habitación leyendo, escribiendo y borrando, a Fede por la paciencia con la que me ha ayudado con el proyecto, a Borja, Pablo y Carla por recibir con entusiasmo mi prototipo, y a Néstor por inspirarme para no quedarme atrás.

Resumen

La creación y la evaluación de corpus narrativos son dos tareas de suma importancia en el campo de la Narratología Computacional. Habitualmente estas tareas son realizadas manualmente por expertos en narración y en el dominio de conocimiento de las historias que forman el corpus, lo que supone un proceso costoso en términos de tiempo y dinero. Debido a este coste, es habitual que las aplicaciones desarrolladas en áreas como la Generación de Historias no dispongan de corpus suficientemente extensos sobre los que trabajar. Este proyecto trata de aprovechar el potencial de las aplicaciones de colaboración masiva de la llamada Web 2.0 para realizar estas tareas, simplificando la creación y evaluación de corpus narrativos en varios aspectos: sin requerir la participación de expertos, sin tener gastos asociados a la contratación de evaluadores, y procesando muchas más historias en menos tiempo. Para este propósito se propone un método de creación y la evaluación de corpus mediante colaboración masiva basado en un juego serio web al que los usuarios acceden voluntariamente para divertirse inventando historias, mejorándolas y dando su opinión sobre las que han creado otros usuarios, mientras que internamente se almacenan las estructuras narrativas generadas y las valoraciones correspondiente de los usuarios. Como caso de estudio se ha desarrollado un minijuego llamado *Zombie Story Maker* que ha sido puesto a disposición del público y cuyos resultados se analizan y discuten al final de este documento. Como conclusión, se exponen las limitaciones que tiene este método en términos de la complejidad de las historias que pueden obtenerse, pero también las facilidades de aumentar la escala de las aplicaciones narrativas que ofrece este método tiene frente a otros más convencionales.

Palabras clave

Creación de Historias, Evaluación de Historias, Estructuras Narrativas, Corpus Narrativos, Colaboración Masiva, Tercerización Masiva, Juegos Serios, Narratología Computacional

Abstract

The creation and evaluation of narrative corpora are two tasks of great importance in the field of Computational Narratology. Usually these tasks are manually performed by experts on narration and the knowledge domain of the stories that form the corpus, what means an expensive process in terms of time and money. Due to this cost, it is usual that applications developed in areas as Story Generation do not have extensive enough corpora for working with. This project try to take advantage of the potential of the mass collaboration applications of the so-called Web 2.0 for performing these tasks, simplifying the creation and the evaluation of narrative corpora in several aspects: without requiring experts' participation, without having expenses associated with the hiring of evaluators, and processing many more stories in less time. For this purpose it is proposed a method for creating and evaluating corpus by mass collaboration based on a web serious game to which users voluntarily access for having fun inventing stories, improving them and giving their opinion about those created by other users, while internally the generated narrative structures and the corresponding assessments from the users are stored. As case study it has been developed a minigame called Zombie Story Maker that has been made available to the public and whose results are analyzed and discussed at the end of this document. As conclusion, the limitations of this method in terms of complexity of the stories that can be obtained are explained, but also the chances of scaling up the narrative applications that this method offers against more conventional ones.

Keywords

Story Creation, Story Evaluation, Narrative Structures, Narrative Corpora, Mass collaboration, Crowdsourcing, Serious Games, Computational Narratology

Índice

Autorización	V
Agradecimientos	IX
Resumen	XI
Abstract	XIII
1. Introducción	1
1.1. Problemas	1
1.2. Objetivos	3
1.3. Estructura del trabajo	4
2. Revisión del estado del arte	5
2.1. Teoría de estructuras narrativas	5
2.2. Generación de historias	10
2.3. Creación y evaluación de corpus narrativos	17
3. Propuesta: Corpus narrativos y colaboración masiva	29
3.1. Requisitos del sistema	29
3.2. Modelo de estructura narrativa	31
3.3. Crear y evaluar corpus mediante colaboración masiva	34
4. Caso de estudio: Zombie Story Maker	39
4.1. Dominio de la aplicación	39
4.2. Estructuras narrativas del subgénero zombi	42
4.3. Implementación mediante tecnologías web	54
5. Resultados y discusión	65
5.1. Análisis del corpus obtenido	65
5.2. Comparación con otros métodos de creación y evaluación	73
6. Conclusiones	77

6.1. Contribuciones	77
6.2. Trabajo futuro	80
A. Zombie Story Maker: Instrucciones de uso	83
A.1. Mensaje enviado a los evaluadores	83
A.2. Mensaje enviado al público en general	85
B. Zombie Story Maker: Corpus obtenido	87
B.1. <i>Carla Carla</i>	87
B.2. <i>Criatura</i>	88
B.3. <i>Desayuno con zombies</i>	90
B.4. <i>Diary of the dead</i>	90
B.5. <i>El pequeño traidor</i>	92
B.6. <i>El zulo maldito</i>	92
B.7. <i>Fui a por el ordenador y me comieron el cerebro</i>	93
B.8. <i>La matanza</i>	94
B.9. <i>Motel cristina</i>	95
B.10. <i>Ocaso</i>	95
B.11. <i>Prueba</i>	97
B.12. <i>Prueba 01</i>	98
B.13. <i>Regustico esquizoide</i>	99
B.14. <i>ZEPS</i>	99
B.15. <i>Zombieland</i>	101
B.16. <i>Zombies, sex and chaos</i>	102
B.17. <i>historia simple</i>	106
B.18. <i>prueba</i>	106
B.19. <i>tret</i>	107
B.20. <i>z!</i>	107
C. Zombie Story Maker: Historia de referencia	111
C.1. Sinopsis de la película Zombieland	112
C.2. Esquema de la estructura narrativa	113
C.3. Formato interno de la estructura narrativa	114
C.4. Representación gráfica de la estructura narrativa	118
Bibliografía	125

Índice de figuras

2.1. Captura de pantalla de la aplicación Sherezade	25
4.1. Personalización de eventos en Zombie Story Maker	43
4.2. La lista de eventos tal y como se muestran en Zombie Story Maker	47
4.3. Pantalla de creación de personajes	48
4.4. Pantalla de creación de lugares	49
4.5. La pantalla de creación de objetos	49
4.6. Un ejemplo de una introducción: se presenta a los personajes y la primera amenaza.	51
4.7. Una «escena» de combate contra zombis del nudo de una his- toria.	54
4.8. Captura de la pantalla principal de la aplicación Zombie Story Maker	55
4.9. Un evento recién añadido. Por supuesto, de momento está vacío.	56
4.10. Si queremos, podemos añadir al evento anterior un lugar... .	56
4.11. Después un objeto. No es obligatorio llenar todos los huecos si no queremos. En este caso queremos especificar cómo han atrincherado el lugar.	57
4.12. Luego decidimos añadir personajes al evento. El orden en el que pongamos cualquier elemento da igual, podíamos haber añadido los personajes al principio si hubiésemos querido. . .	57
4.13. Como queremos darle a este evento un toque más personal, añadimos un texto personalizado.	57
4.14. Los menús de Personajes, Objetos y Lugares, tal como quedan al desplegarlos en nuestro ejemplo. Pueden observarse los dis- tintos Elementos creados.	58
4.15. Menú principal de nuestra aplicación.	61
4.16. Ejemplo de la pantalla de resumen de una historia.	62
5.1. Gráfico comparativo entre el número de historias completas, incompletas y vacías	66

5.2. Gráfico comparativo entre la proporción de contenido textual entre historias completas e incompletas	67
5.3. Gráfico comparativo entre los elementos narrativos de las historias completas e incompletas	69
5.4. Captura de pantalla de la aplicación Sherezade mostrando la línea temporal	74
5.5. Captura de pantalla de la aplicación Sherezade, mostrando eventos en la línea temporal	76
6.1. Datos sobre el número de visitas de usuarios distintos a la aplicación.	78
C.1. Lista de Personajes de nuestra historia	119
C.2. Lista de Objetos de nuestra historia	119
C.3. Lista de Lugares de nuestra historia	120
C.4. Lista de Eventos de nuestra historia (1)	121
C.5. Lista de Eventos de nuestra historia (2)	122
C.6. Lista de Eventos de nuestra historia (3)	123
C.7. Lista de Eventos de nuestra historia (4)	124

Índice de tablas

5.1. Tabla de puntuaciones de las historias	69
5.2. Tabla con información estructural de las historias creadas en la aplicación	71
5.3. Tabla con información sobre la valoración y el uso de las historias creadas en la aplicación	71

Capítulo 1

Introducción

RESUMEN: En este capítulo se introducen los conceptos básicos sobre la narratología computacional y se habla en particular sobre los editores y anotadores narrativos y los problemas específicos que acarrearán. Tras ello, presentamos nuestra propuesta y objetivos como un modelo de edición de corpus narrativos colaborativo on line y se describe la estructura posterior que seguirá el trabajo.

1.1. Problemas

La narración automática es el proceso de generar y comunicar de alguna forma una historia mediante un sistema informático. Obviamente las historias generadas no deben existir previamente, al menos en su forma final, y deben cumplir además con un mínimo de coherencia y originalidad. Las aplicaciones de generación de historias comenzaron a estudiarse en los 70 y durante dos décadas se realizaron numerosos trabajos sobre ella, dando como resultado diversas aplicaciones que creaban historias con mayor o menor éxito. Después, tras largo un periodo de estancamiento, principalmente debido al estancamiento de los paradigmas de inteligencia artificial, recientemente ha vuelto a resurgir, gracias al gran avance de las tecnologías en los ámbitos en los que la narración automática puede ser aplicada, como por ejemplo el de los videojuegos o las aplicaciones multimedia. Sin embargo, a pesar de la potencia de las nuevas aplicaciones de narración desarrolladas, el mundo de la narración automática se ha encontrado con un nuevo cuello de botella, que esta vez no es técnico: la carencia de un corpus amplio sobre el que trabajar (Gervás, 2010). La narratología computacional requiere de grandes corpus de historias para poder empezar a desarrollarse como ciencia, pero esos corpus son muy costosos de construir y las herramientas que existen para construirlos son muy costosas y complejas de usar. No sólo eso, sino

que uno de los problemas a los que se enfrentan los investigadores es que sus herramientas normalmente parten de modelos o paradigmas diferentes y por tanto deben partir de cero a la hora de desarrollar una base de historias que puedan utilizar para evaluar o contrastar sus teorías, pues lo que cada corpus contendrá información distinta o almacenada de forma diferente, de acuerdo con el objetivo de cada paradigma. Ya sean herramientas para editar o anotar historias o generadores automáticos de narrativa, es necesario tener o bien una buena cantidad de modelos de historia que la herramienta pueda leer, interpretar o usar o bien un grupo de usuarios dispuestos a crearlas. Además de este problema endémico de cualquier herramienta que tenga que trabajar sobre un corpus, tenemos otro adicional: la evaluación de las historias.

Tradicionalmente, el modo de evaluación de historias siempre ha sido un quebradero de cabeza para los investigadores y recientemente se están consiguiendo algunos avances en la metodología y las métricas usadas (Corradini et al., 2009); pero aún sigue siendo algo impreciso y complejo, sobre todo en cuanto a valorar la coherencia y emoción de una historia. Algunos investigadores han propuesto que la evaluación pueda hacerse mediante anotación de las historias, dado que las funciones automáticas tienen problemas a la hora de manejar conceptos como la moraleja de una historia o el interés (Kwong, 2010). Por otro lado, en nuestro caso junto con la evaluación de las historias tenemos otra tarea similar: la anotación de los corpus. Decimos que es una tarea similar porque exige especificar aspectos no triviales o explícitos de la historia (en mayor o menor medida, dependiendo del tipo de anotación) tales como las motivaciones de los personajes, los puntos en donde se da énfasis o los efectos narrativos del tipo «al borde del abismo» o *cliffhanger*. Al hecho de que de nuevo es una tarea difícilmente automatizable o formalizable, aunque hay estudios sobre ello (Chambers y Jurafsky, 2010), le debemos sumar que por lo general el proceso de anotación no suele ser llevado a cabo por una sola fuente, con lo que se debe solventar el problema conocido como el acuerdo entre anotadores.

Por ello, recientemente han surgido un buen número de trabajos cuyo objetivo es generar amplios corpus narrativos que incluyan descripción de referencias, información sobre emociones, información sobre la semántica subyacente al discurso, relaciones no triviales, modos de narración, líneas temporales, etc. Y una de las últimas vías abiertas en el campo (Swanson y Gordon, 2008) es la del uso de internet para facilitar la ficción colaborativa. Esto quiere decir que la web permite a números grandes de personas contribuir a una misma obra, como en las «wikinovelas» (Wikinovela, 2006) o «wikipelículas», sino que podemos también usarla como base de conocimiento. Así, recientes trabajos dejan entrever que se puede aprovechar esta clase de sabiduría colectiva emergente (Galton, 1907) para la creación de historias. No sólo eso, sino que también se puede reusar el conocimiento ya

existente de alguna manera y almacenarlo para construir corpus, tarea que de hacerse a partir de 0 y a mano, requeriría una enorme cantidad de tiempo y gente. Así, sobre esta base de conocimiento ya existente, se podrían luego realizar tareas de búsqueda, comparación o anotación, solucionando parte del problema de la ausencia de corpus usables, pero apareciendo un nuevo problema: ¿Qué técnicas y estrategias usar y qué conocimiento extraer de la red?

En resumen, la narrativa computacional se enfrenta actualmente a una nueva era de retos, tras haber superado los antiguos problemas derivados de las limitaciones técnicas o los paradigmas, dando como resultado una gran cantidad de estudios actuales sobre estos principales problemas, como lo son la creación, anotación o evaluación de historias y corpus narrativos.

1.2. Objetivos

¿Qué motivos hay detrás de este trabajo? En primer lugar, al revisar y analizar los distintos editores de historias o sistemas de narración, hemos llegado a la conclusión de que todos son excesivamente complicados de usar. No sólo eso, sino que salvo excepciones están relegados a un uso exclusivo en el mundo de la investigación. Una de las metas que se buscan con este trabajo es proponer un sistema asequible para usuarios medios, pero que mantuviese el máximo posible de funcionalidad narrativa. Eso hace que, como decíamos en la introducción tengamos una gran diversidad de herramientas de edición narrativa, tanto interactiva como clásica, pero no haya una gran base de historias o de autores susceptibles de querer usarlas.

Por último, no sólo es necesario que la historia pueda extraerse de forma intuitiva a partir de ver el modelo, sino que muchos editores salvo excepciones obvian otro aspecto importante, que es el mismo diseño de la interfaz para que sea usable: muchas veces los elementos entidades o conceptos de la historia están en pantallas distintas a la del guión de la misma, o incluso para usar o crear un concepto hay que ir a otra pantalla. Además faltan muchas veces elementos que ya están en el paradigma de uso de las aplicaciones, como el drag & drop. La razón de esto suele ser que el objeto de estudio que están haciendo no requiere de hacerlo usable, sino que se centran en el modelo o en la generación. Aunque en lo que corresponde al presente trabajo, tampoco sea un objetivo principal el desarrollo o estudio de una interfaz para editor, sí que queremos lograr que el prototipo fuese lo más intuitivo posible, porque no sólo tiene un efecto cosmético o usable, sino que nos va a permitir aumentar el dicho parámetro de la visibilidad de forma indirecta: en nuestra opinión no sólo es importante que de un solo vistazo al modelo narrativo de una historia se pueda construir mentalmente la misma, sino que también es crucial que se vea enseguida qué puedes hacer en el editor y qué no, y qué elementos hay disponibles y cuáles no.

En resumen, presentaremos un modelo y también una herramienta que intente que la labor de edición sea más sencilla mientras se mantiene lo más posible la completitud del modelo narrativo. Sobre dicho modelo, hemos decidido centrarnos en un subgénero narrativo concreto. Esto nos permitirá garantizar que nuestro modelo creará siempre historias completas al tiempo que es lo suficientemente sencillo de usar como para mantener el compromiso de interfaz que establecíamos antes. Una vez conseguido esto, estudiaremos la posibilidad de usar nuestro modelo como herramienta colaborativa en internet, poniéndolo a disposición de cualquiera, para comprobar el efecto que puede tener ello en la construcción y anotación de historias.

1.3. Estructura del trabajo

El presente trabajo está estructurado de tal forma que en el siguiente apartado revisaremos ampliamente diversas teorías semánticas y narrativas, comentando que aspectos de cada una pueden ser útiles o de consideración, para a continuación repasar los ejemplos más representativos de las herramientas de edición narrativa que han ido surgiendo a lo largo de la historia y sus características. Después, en el tercer capítulo hablaremos del modelo propuesto y la teoría de computación colaborativa que queremos poner en práctica, comentando recientes ejemplos prácticos y cómo la adaptaremos a nuestro modelo.

Después, en el capítulo cuarto, describiremos a fondo la herramienta prototipo que hemos desarrollado como demostración de nuestro modelo. También hablaremos de por qué hemos escogido el dominio que usamos con nuestro prototipo escogido y de sus características únicas y otras más comunes. Una vez descrito nuestro modelo de ejemplo y la aplicación, pasaremos en el quinto capítulo a compararla con otras aplicaciones de edición y anotación de historias, y mostrar la evaluación práctica que hemos realizado sobre ella, junto con varios datos comparativos de los resultados. Finalmente, en el sexto capítulo expondremos nuestras conclusiones y revisaremos los caminos que abren a un posible trabajo futuro.

Capítulo 2

Revisión del estado del arte

RESUMEN: En este capítulo se presentan las bases sobre las que hemos inspirado nuestro trabajo, describiendo primero algunas teorías narrativas o de construcción de lenguaje que nos han resultado inspiradoras para a continuación analizar el la escena de las herramientas d enarración, contrastando distintos sistemas, tanto antiguos como modernos. Por último, hacemos una completa revisión de las aplicaciones para edición narrativa , mostrando sus características distintivas y los objetivos a los que aspiran, así como la necesidad de disponer de un corpus narrativo abundante y de un editor sencillo e intuitivo.

2.1. Teoría de estructuras narrativas

La Narratología, término acuñado en francés (Todorov, 1973), es una disciplina teórica que estudia todas las clases de narrativas con el objetivo principal de crear una descripción profunda de la estructura de las mismas. Con estructura profunda podemos referirnos a descomponer la narrativa en sus distintos componentes y deducir sus funciones, propósitos y relaciones entre ellas. Generalmente una narrativa cumple las siguientes características:

- Contiene al menos una secuencia temporal de eventos.
- Estos eventos además suceden en una localización (por tanto los eventos serán espaciotemporalmente ubicables).
- Debe haber una coherencia semántica general, o un «tema». Esta característica es difícil de formalizar, pero distingue a una historia de una lista de sucesos inconexos. Mas adelante hablaremos de conceptos como moraleja de la historia o «story point» dada su relevancia para nuestro estudio y el interés que suscitan entre los investigadores.

- En los eventos de la historia, habrá alguna clase de actores con una serie de características definidas que además realizan o reciben acciones. En pocas palabras, tienen un papel en la historia.

Sin embargo, aunque esta sea una lista general de las características de una narrativa, siempre pueden surgir variaciones o historias experimentales, por lo que conviene (a nosotros y en general a todo investigador de este campo) centrarse en alguna clase de narrativa concreta o aspecto específico. De hecho, han surgido muchas clases de narratologías especializadas, por ejemplo la narratología psicoanalítica, la socioeconómica, la feminista o la cognitiva, cuyos dominios acotados les han permitido a los investigadores obtener buenos resultados.

A la hora de poner en práctica las teorías que trabajan con estas narrativas, también se pueden distinguir distintos grupos según su objetivo o utilidad. Por ejemplo, Zarri (Schilder, 2010) divide el conjunto de todas estas técnicas en dos dependiendo de su finalidad: las puramente narrativas y las crónicas electrónicas (*e-Chronicles*).

La Narrativa se refiere al estudio de las diferentes técnicas de construir historias y eventos generando palabras, imágenes y sonidos con el objetivo de entretener, describir o enseñar. Una subrama de este grupo sería la narrativa digital, que concierne al dominio de la computación multimedia y sus formas específicas de utilizar personajes, emociones y otros elementos. Michael Mateas va más allá y separa un subconjunto de ésta: la inteligencia narrativa, que explora las relaciones entre la inteligencia artificial, la expresión multimedia y la interfaz humano-máquina.

Por otro lado, un sistema de crónicas electrónicas se define como una forma de analizar, almacenar y organizar una serie de fuentes de eventos multimedia obtenidos por un tercero. Las crónicas almacenadas pueden ser de cualquier tipo: resúmenes, transcripciones de discursos, videos, registros, señales de audio, canciones, etc. Los objetivos de estas disciplinas son obtener una base de conocimiento coherente utilizando todos estos materiales en bruto y permitir un acceso específico según desee un usuario. Este tipo de técnicas tienen mucho que ver con los anotadores, los buscadores semánticos, recomendadores, etc. En este trabajo, nos centraremos en el primer subgrupo, en concreto en la narrativa digital y dentro de ella, los editores de historias narrativas. En este apartado comenzaremos analizando varias teorías de proceso de lenguaje, entendiendo que son necesarias en tanto que también estudian una relación de eventos dentro de un posible texto narrativo, y después veremos una breve revisión de diversas técnicas narrativas utilizadas a lo largo de la historia, y a continuación, para afinar más hacia nuestro objetivo haremos una descripción de diversas herramientas de edición o generación y su aportación al estado del arte, describiendo sus ventajas y también los problemas con los que se encontraron o defectos que tienen.

Se conoce como procesamiento semántico del lenguaje al análisis y generación de las ideas concretas que transmite un discurso. Puesto de manera simple, se centra en el «qué quiere decir», en lugar del «cómo decirlo». Por tanto, inicialmente descartaremos el aspecto sintáctico del lenguaje, aunque trataremos más adelante en este trabajo la relación entre este aspecto y el semántico. La razón de que nos hayamos fijado en este tipo de procesamiento, es que pueden relacionarse de forma intuitiva con la construcción de una historia, aunque en su caso, estas simplemente construyen cualquier clase de texto. Al igual que para construir lenguaje los seres humanos crean relaciones de causalidad y mantienen una ontología subyacente de los elementos de un texto en lenguaje natural, ya sea hablado o escrito, en una historia podemos tener una idea intuitiva de las relaciones entre los sucesos que en ella ocurren y extraer información no trivial. Por tanto, parece interesante que nos fijemos en que metodologías y modelos usan algunas teorías que trabajan con la semántica del lenguaje para ver si nosotros podemos inspirarnos en ellas.

En el mundo académico han surgido diversas teorías sobre la semántica del lenguaje, por lo general creadas desde el punto de vista psicológico-lingüístico. La teoría del metalenguaje semántico natural (NSM) es una de ellas. Esta teoría fue propuesta en los 70 por Anna Wierzbicka (Wierzbicka, 1972), y coge fuerza a finales de los 90 (Goddard, 1998). Se basa en la noción de construir oraciones usando conceptos semánticos. Estos conceptos, conforman los componentes atómicos de cualquier lenguaje: las primitivas semánticas. El concepto intuitivo de esta teoría es que si una persona tuviese que definir todos los términos contenidos en el diccionario de un idioma, rápidamente se encontraría con ciclos: términos que se definen con otros que a su vez se definen con los primeros, con lo que la teoría indica que ciertos conceptos son irreductibles, y simplemente se definen consigo mismos. Sin embargo, esta teoría es muy difícil llevarla a la práctica puesto que no da un método formal para extraer estos conceptos, sino que es muy ambigua y plantea serios problemas a la hora de extraer estos conceptos dependiendo del idioma o de lo que consideremos como «definición». No es de extrañar que esta teoría no haya pasado del campo puramente teórico o de modelos sobre dominios muy reducidos.

Si la teoría del NSM permitía teóricamente desmenuzar un lenguaje en componentes atómicos y significativos, solo nos faltaría obtener una teoría que trabajase con ellos; pues bien, en la misma década en la que surgía el NSM, a partir de la lógica borrosa (un campo en auge por entonces, aunque prácticamente teórico), surgió la cuestión de si podrían aplicarse directamente sus principios al razonamiento con lenguaje natural. Esta

Tradicionalmente, la computación exigía trabajar con números y símbolos, usados por los algoritmos de razonamiento como representaciones lógico-matemáticas. Sin embargo, los humanos emplean palabras del lenguaje natural para razonar, extrayendo conclusiones de ellas, también expresadas con

palabras. Estas palabras además tienen denotaciones difusas: en el ejemplo del termostato nos surgía el concepto de «temperatura alta» que no tiene una definición concreta, sino que se deja a la intuición de cada uno. Para poder llevar esto al razonamiento máquina se pueden usar técnicas de lógica borrosa, y particularmente este campo actualmente inexplorado de la misma conocido como «Procesamiento con palabras» («Computing with words» o CW), una rama de la lógica borrosa comenzada por Lofti Zadeh en el 72 (Zadeh, 1999).

Lamentablemente, tanto la teoría de Zadeh como la de Wierzbicka se han mantenido en el campo teórico, como hemos dicho, pero contienen conceptos muy interesantes, de los que nos hemos inspirado. Concretamente el concepto de «gránulo semántico» (connotación semántica de una palabra) del procesamiento con palabras es inspiración directa de nuestros eventos semánticos.

Sin embargo, en esa época surge otra teoría que si ha tenido intentos prácticos con un éxito relativo aunque no determinante. Hablamos de la teoría de la dependencia conceptual de Roger Schank (Lytinen, 1992), que surge en los 70 con esta aproximación semántica. La teoría de Schank incluía una metodología mucho más detallada, y una serie de aplicaciones prácticas claras que incluían razonamiento basado en casos, lógica de predicados o teoría del conocimiento.

Inicialmente, Schank comienza definiendo las primitivas más bien como relaciones conceptuales, independientemente de la clasificación y ontología de los conceptos. Así, define las siguientes relaciones:

- PTRANS: Transferencia de lugar de un objeto.
- ATRANS: Transferencia de posesión de un objeto.
- MTRANS: Transferencia de información mental entre agentes.
- MBUILD: Construcción de pensamiento o información mental por un agente.
- ATTEND: Dirigir la atención de un órgano sensitivo hacia un objeto.
- GRASP: Tomar un objeto.
- PROPEL: Aplicar fuerza física a un objeto.
- MOVE: Movimiento de una parte de un objeto hecha por ese mismo objeto.
- INGEST: La introducción de un objeto dentro de un agente hecha por ese mismo agente.
- EXPEL: Expulsión de un objeto fuera de un agente hecha por ese mismo agente.

- SPEAK: Producir sonido, sea comunicativo o no.

Además de las relaciones, los conceptos pertenecerán a una serie de categorías, pues como hemos visto, algunas de las relaciones necesitan de objetos animados, mientras que otras, necesitan de objetos físicos, pudiendo llegar a ser tan específicos incluso que otras relaciones necesitarán de humanos. Para definir que conceptos necesita para poder ser expresada, cada una de estas relaciones tiene una o varias plantillas. Por ejemplo, la dependencia PTRANS tiene la siguiente plantilla:

- ACTOR: un objeto animado que inicia la acción.
- OBJECT: un objeto físico que recibe la acción.
- FROM: un lugar, desde el que se comienza la acción.
- TO: un lugar, en el que acaba la acción.

Así, cuando interpretásemos una frase en la que el verbo perteneciese a la categoría de PTRANS, sería necesario buscar los candidatos para cada uno de estos papeles, ya sea en esa frase o en otras o incluso dejarlos implícitos. Como el lector podrá adivinar, identificar que concepto encaja en cada papel, es necesario alguna clase de análisis sintáctico. Schank no obvia esto, sino que indica que es incluso deseable que este análisis esté unido a la extracción de primitivas de una frase. El concepto de las plantillas de Schank ha sido muy recurrido en muchos otros estudios e incluso nosotros nos hemos inspirado en él para elaborar nuestros eventos semánticos, de los que hablaremos en el capítulo siguiente.

También esta teoría intenta solventar los problemas de ambigüedad de términos de forma semántica: aplicando lo que llama como guiones, sitúa las relaciones en un contexto concreto de acciones posibles. De hecho, extendiendo estos guiones, o scripts, puede construir otra especie de plantillas, que pueden generar todos los guiones de un mismo tipo. De esta forma, una plantilla que generalizase «obtener un servicio comercial» podría convertirse en el guión de «comer en un restaurante» o en el de «ir al médico» si se rellenan las acciones correctas en los huecos que permite.

Schank posteriormente llega un paso más allá introduciendo objetivos y planificación. Gracias a ellos, se podría construir un sistema de razonamiento que construyese las acciones necesarias para conseguir un objetivo. Estas acciones estarían en la forma de representaciones semánticas, así que podrían ser usadas para generar lenguaje pero también para procesarlas y llevar a cabo acciones. La teoría de Schank lleva más allá el uso de primitivas semánticas y además describe una metodología clara y construcciones modulares para su uso. Está claramente orientada a su aplicación práctica en computación y de hecho su equipo de investigación realizó varios trabajos

prácticos, aunque de momento el progreso de este campo de investigación haya quedado detenido.

Como vemos, a lo largo de la historia reciente de la narrativa, han surgido diversas teorías que intentaban manejar el lenguaje de forma semántica, aunque no han tenido mucho seguimiento práctico. Sin embargo, veremos que los conceptos que postulan no están muy alejados de los estudios modernos sobre narrativa.

2.2. Generación de historias

Inicialmente, las aplicaciones sobre este campo se prodigaron de forma numerosa, en la década de los 70 y 80, pero fueron decayendo poco a poco principalmente debido a un estancamiento de las técnicas que usaban. Sin embargo, recientemente y debido a nuevos avances en el campo de la inteligencia artificial y a la nueva orientación que está tomando la industria del entretenimiento, hay un renovado interés por parte de investigadores y empresas por desarrollar herramientas prácticas que generen historias.

Principalmente, las aplicaciones de generación de historias se han encontrado con tres grandes problemas de forma reiterada (Peinado, 2008):

1º El problema del dominio

Dado que las aplicaciones de narración son aplicaciones que trabajan con conocimiento, heredan el principal problema de los sistemas de conocimiento: el principal cuello de botella es cómo y qué conocimiento adquirir, almacenar y utilizar. Las aplicaciones narrativas tienen otro gran problema: mientras que las aplicaciones comunes basadas en conocimientos pueden permitirse formalizar y limitar mucho su dominio, los requisitos iniciales de las primeras son tan ambiciosos como para utilizar el dominio del lenguaje natural o del «sentido común», ambos demasiado bastos o ambiguos como para que sea trivial modelizarlos. Así pues, tenemos un gran problema a la hora de definir el dominio con el que trabajará la aplicación y a lo largo de la historia las aplicaciones narrativas han limitado su dominio a un coto cerrado para poder mostrar resultados convincentes.

2º El problema del contraste

Una característica curiosa de las aplicaciones que han surgido (y que iremos viendo a continuación) es que por lo general no se inspiran en aplicaciones anteriores, o extienden teorías comenzadas en aquellas; en su lugar, se suele dar una propuesta nueva, más acorde con lo que piensa el investigador que debería ser el camino a seguir. Esto provoca que tengamos una gran cantidad de aplicaciones basadas en tecnologías y teorías muy dispares, tengan restricciones cualitativamente distintas o ataquen problemas diferentes. Por ello, es casi inútil compararlas entre sí y extraer conclusiones mejores sobre las técnicas que usan.

3º El problema de la evaluación

Por último, al ver los distintos trabajos de narración interactiva, vemos que los métodos de evaluación son heterodoxos como mínimo: por lo general no usan métricas ni se comparan con otros trabajos (como decíamos en el punto anterior), llegando algunos casos a carecer por completo del método de evaluación. Por otra parte, evaluar una historia no es algo trivial: cómo debería evaluarse si una aplicación de narración narra correctamente? Usualmente se suele usar algún tipo de métrica adhoc para medir la creatividad y la coherencia, pero al ser términos tan nebulosos estos últimos, los resultados son cuestionables. Otra manera popular de medir la calidad de las historias son las encuestas a personas: en éstas se miden parámetros como los antes mencionados, pero normalmente no se hacen con métodos estadísticos rigurosos y por tanto entra en juego a menudo la subjetividad de los sujetos encuestados.

Además de estos problemas, podríamos añadir otros menores, como que intentar atacar distintos tipos de narración interactiva requiere de distintas aproximaciones, o que los distintos medios a través de los que se puede narrar (audio, video, texto, etc.) tienen cada uno características especiales propias.

Para mostrar el avance que ha experimentado este campo, a continuación revisaremos las principales aplicaciones que han ido surgiendo con los años en el campo de la generación de historias.

El primer trabajo sobre generación de historias producto de investigación fue Automatic Novel Writer (Klein et al., 1973), que podía construir historias largas de misterio y se considera un sólido punto de partida de este campo de investigación. La generación de historias realizada por Automatic Novel Writer estaba basada en reglas, y permitía que el usuario introdujese inicialmente algunos parámetros iniciales. Aunque los resultados de este proyecto eran un poco «robóticos», las historias eran coherentes e incluso contaban hechos que ocurrían en base a una causalidad implícita.

Vemos que este primer producto ya tenía como objetivo la que sería la meta común de todas las aplicaciones que irían surgiendo después: conseguir generar historias completas (un objetivo que suena engañosamente simple). Inicialmente, las aplicaciones que se desarrollaban se ceñían a intentar alcanzar este objetivo y no fue hasta entrados los años 90 que comenzaron a experimentar con otras orientaciones.

Así, Minstrel (Turner, 1992) es una aplicación cuyo objetivo era generar historias creativas, e intentaba imitar los procesos mentales que realizamos los humanos para inventar cuentos usando redes semánticas. En este caso, el dominio estaba limitado a historias Artúricas, y el conocimiento estaba representado en estructuras al estilo lisp. Minstrel contenía un motor de razonamiento basado en casos y eso le permitía crear nuevas historias a partir de un corpus de historias ya guardadas, permitiendo deducir que era lo que debía pasar a continuación de un suceso o episodio dado de la historia. Además, permitía usar un sistema de objetivos y acciones para los person-

ajes e incluso para el autor. Sin embargo, Minstrel tenía algunos problemas de formalismo con su modelo de conocimiento dado que era un tanto libre y poco definido: los conceptos se almacenaban y clasificaban de la misma forma que las entidades individuales, no existía tipado de roles, y usaba relaciones propietarias un tanto ad hoc. Posteriormente se hizo un esfuerzo de adaptar este modelo a una ontología más estandarizada en OWL, que permitía una validación automática y una edición más sencilla, sobre todo porque no implicaba conocer estructuras con una sintaxis como la de lisp. Lamentablemente sólo se quedó en transformar el modelo de conocimiento, y no se extendió al motor completo (Peinado y Gervás, 2006).

Por otro lado, Brutus (Bringsjord y Ferrucci, 1999) era un proyecto que realizaba versiones distintas de una misma historia creada por un humano inicialmente, y de acuerdo a unos parámetros que el mismo seleccionaría. Los autores de Brutus hicieron esto en contrapunto al sostenimiento de la idea de que no se podría replicar la creatividad humana, pero sí que se podía coger una historia y obtener algo parecido creando una nueva (basada en la misma). Brutus era capaz de generar historias increíblemente realistas, con una estructura muy clara y una buena capacidad para la descripción. Utiliza algunos recursos interesantes además, aunque a veces podía generar una conclusión un poco ilógica. Quizá esto es debido a que el dominio era estricto: brutus generaba historias sobre traición siempre, variando los demás aspectos.

Otros proyectos se centraban en la forma de la narración: SWAN (Lu y Zhang, 2002) tenía como objetivo hacer un estudio de la narración multimodal. Como mencionamos antes, una característica común es que la narración de una historia puede realizarse a través de diferentes medios, cada uno con características propias. De esta forma, el texto escrito puede requerir de unas estructuras narrativas completamente distintas a la imagen en movimiento, a una historia contada. Podría parecerse que los recursos usados en uno u otro medio tienen siempre algún equivalente en los demás pero esto no es algo trivial de demostrar y de momento se tratan de forma distinta. En el caso de SWAN se generaban a partir de una descripción realizada por el usuario historias narradas mediante animaciones.

En cuanto a la teoría en la que se basan las aplicaciones la mayoría utilizan tecnología inspirada en teorías clásicas como la de Propp usada en aplicaciones como Joseph (Lang, 1997) o la de Schank, usada principalmente en la aplicación que desarrolló: Tale Spin (Meehan, 1981), en gramáticas generadoras de historias o incluso en enormes compendios de reglas de causalidad para ser usados por motores de razonamiento. En el caso de Tale Spin, se usaban con éxito plantillas para las historias, pero muchas veces se introducían hechos o frases obvios o sin importancia, generando historias algo aburridas. Joseph sin embargo usaba reglas gramáticas para construir la historia, pero las historias que generaba a veces resultaban un tanto extrañas, al no estable-

cer causalidad, sino eventos al estilo de acción-reacción. Sin embargo, tenía una característica interesante, pues permitía establecer la reacción emocional de su protagonista a ciertos eventos y a su vez, establecer acciones que eran ejecutadas según el estado emocional del mismo.

La separación en niveles cualitativos que puede tener una historia también depende del teórico en cuestión, habiendo trabajos muy divergentes en este aspecto, aunque lo habitual es tener un nivel narrativo o de historia, que es el que contiene todo lo que ocurre en la historia, y un nivel de discurso, que simplemente contiene los símbolos o los eventos que se van a contar (ya sea simplemente el texto, o incluso más conocimiento). Otros investigadores, como Peinado o Bal (Bal, 1997) definen un tercer nivel conocido como «nivel de fábula», en el que pueden cambiar los valores abstractos y el sentido que toman los acontecimientos: en la historia (nivel de historia o simulación) puede suceder una cosa, y no sólo puedes elegir contarla de distintas maneras (nivel de discurso), sino que puedes valorar lo que sucede de distinta forma (nivel de fábula).

También se pueden tomar posturas intermedias o niveles narrativos adicionales. Por ejemplo, Nick Monfort (Monfort, 2007) utiliza el concepto de mundo focalizado en torno a un personaje, para que una vez seleccionado el personaje en torno al que se enfocará la narración, no sólo contaría lo que le sucedía a dicho personaje, sino que su herramienta generaba todo un mundo «parcial» a partir del mundo completo del nivel de historia con todos los sucesos que el personaje conozca, y una vez obtenido, se obtendrá el discurso a partir de él. Peinado también introduce un concepto intermedio, el nivel de fábula, en el que se encuentran los hechos una vez interpretados por la narración (éticamente, emocionalmente, tomando un punto de vista, o desconociendo ciertos eventos que hayan ocurrido), pero antes de ser transformados por el discurso. La herramienta NKRL de Zarri (Schilder, 2010), también tiene en cuenta el nivel de fábula, pero lo subsume dentro del nivel de historia. La herramienta de Zarri, analizando diversas técnicas de narratología, reúne una completa teoría sobre representación y gestión de contenido narrativo y desarrolló la herramienta NKRL, con la que permite tratar con lo que el llama «no ficción de relevancia económica». El cometido de NKRL se acerca a la definición que el propio Zarri daba de crónicas electrónicas, esto es, una herramienta capaz de almacenar y clasificar conocimiento y proporcionar resultados a las consultas semánticas de los usuarios. Sin embargo, NKRL además incorpora elementos de razonamiento y representación. Además, acotar el dominio de su herramienta a la narrativa de no ficción es en teoría sólo para enfatizar el aspecto práctico de la misma, no teniendo que ver con la estructura o modelo que usa. El sistema utiliza el concepto de «evento elemental», considerando tal y como hemos definido en este trabajo, que una narrativa es esencialmente un conjunto de eventos relacionados entre sí espaciotemporalmente, pero formalizando el concepto

de evento (que es intuitivo) como una instanciación de una plantilla establecida que identifica la categoría del evento y que admite ciertas propiedades, algunas generales y que tendrán todos, como su situación espacial y temporal y otras más específicas a ese evento, como actores, receptores, etc. Este concepto de plantillas no nos es desconocido, siendo similar al que usaba Schank pero a mucho más bajo nivel: estas plantillas son los elementos atómicos de la narración. Cada una de estas plantillas elementales pertenece además a un grupo definido por su predicado profundo, esto es, la categoría conceptual a la que pertenece. Por ejemplo, si un evento elemental es «Bruto apuñaló a César», éste pertenecería a la plantilla elemental «Herir a Humano» que a su vez pertenece al predicado profundo «Producir», porque es un evento que produce un resultado o un cambio en el estado de uno de los actores. Además, las plantillas como siempre admitirán una serie de propiedades adicionales, como actores participantes y restricciones sobre su tipo. En el ejemplo, la plantilla «Herir a Humano» admite un sujeto, que sería Bruto, una modalidad que sería «apuñalar» y un receptor, que sería César. Además la plantilla tendría también como propiedades la localización Roma y el tiempo 44 AC.

Este concepto de evento como elemento atómico de una historia narrativa se repite a lo largo de casi todas las teorías, aunque cambie de forma. Si en NKRL veíamos que era representado por plantillas de granularidad muy fina, otros como KIM (Kim, 1996) lo hacen con tripletes que llamará eventos-mónada. Estos tripletes contienen un objeto o actor, una propiedad o una relación y un instante. Las propiedades y relaciones serán totalmente libres, permitiendo una generalidad máxima a la hora de diseñar. Sin embargo, esto también quiere decir que para cada suceso que ocurra en un cierto tiempo, debe haber al menos una propiedad en cada actor, por lo que la granularidad es incluso más fina con esta teoría. No es difícil imaginar que para cada objeto de la narrativa en cada instante debemos modelar tantas propiedades como eventos o relaciones tenga, así que la cantidad de eventos-mónada se multiplica de forma geométrica. Además tiene un problema adicional, que es que no da una norma formal que especifique que propiedades deben ser consideradas para construir estos eventos, por lo que hay un alto grado de dependencia del creador de la historia.

Un último ejemplo, interesante porque se aproxima al concepto tanto de Schank como el de Zadeh para los eventos semánticos, es el de Davidson (Davidson, 1967), que usa los verbos como evento pero además les asocia con una serie de variables, y que además permite modificadores. Además, la manera en la que representa los eventos es una forma matemática, utilizando implicaciones y conjunciones. Huelga decir que la gran ventaja de este modelo es que está completamente formalizado, pero por otro lado tiene el mismo problema que vimos con Schank, Zadeh y Zarri: las propiedades o funciones que acepta cada evento se hacen adhoc, así que o tenemos una lista innumerable de eventos con otras tantas relaciones entre ellos, o debemos acotar

nuestro dominio.

Como conclusión, podemos decir que los eventos mantienen una posición prevalente como constitutores de la historia, pero ahora bien, una vez que tenemos estos elementos atómicos, las distintas aplicaciones y teorías narrativas divergen en como usarlos para construir una historia; una historia que además sea narrativa y por tanto tenga una coherencia y una suerte de moraleja. En general, podríamos dividir estas filosofías de generación de historias en dos grupos, que serían la versión de las estrategias «top-down» y «bottom-up» de la narración: la primera toma como referencia el objetivo de la historia, haciendo que todo se construya en torno a un objetivo (y de ahí que muchas aplicaciones que usan esta filosofía utilicen planificación) y la segunda tendría como referencia a los personajes, dejándoles que «actúen» en la historia provocando consecuencias para sus acciones (necesitando de alguna forma algún tipo de razonador de causalidad). Un ejemplo de esta última sería Universe, que se centra principalmente en los conflictos entre personajes. Sin embargo, estas filosofías no son disjuntas, y se pueden utilizar de manera conjunta, como ya se hizo en Rald. La manera de generar la historia es paralela también a la aproximación que realizan los sistemas de narración, que suelen decidirse por centrarse en la trama o por centrarse en los personajes. Si estas dos aproximaciones son isomorfas o no con respecto a las dos estrategias del sistema de narración sería debate de otro estudio más profundo, y sólo diremos aquí que por lo general, en todos estos sistemas, los que hacen estrategia «top-down» se centran en la trama, mientras que los otros lo hacen en los personajes. Partiendo de estas filosofías, se pueden ir haciendo variaciones que les añadan diferentes «sabores», y permitiendo alcanzar nuevas características de personalización o mejoras en la calidad de la historia. Por ejemplo, Rumelhart desarrollaba una gramática en la que a partir de reglas «semánticas», generaba la historia. Estas reglas describían alguna clase de relación entre dos componentes de la historia, y dado una serie de objetivos y personajes, su modelo generaba una historia de forma declarativa. Las reglas semánticas a aplicar se decidían según la estructura sintáctica que pudiese tomar cada trozo concreto de la historia. Usando otro modo declarativo en su herramienta Joseph, Raymond Lang (Lang, 1999) podía generar también historias Proppianas a partir de reglas, en este caso alejándose de la concepción semántica y siguiendo un sistema de restricciones más formal que Rumelhart que extraía directamente de los patrones de las historias del dominio de Propp las restricciones necesarias para permitir un proceso computacional de las relaciones semánticas entre los sucesos de la historia. También permitía procesar predicados temporales. Sin embargo, tenía algunos defectos como el de tener que modelar a reacción y la emoción del protagonista para todos los eventos que le ocurriesen (aunque esto permitía que las emociones del personaje tuviesen impacto en la historia) y que además, sólo permitía un protagonista. Joseph incluía también una suerte de

sistema de planificación de objetivos, aunque no muy completo: no detectaba objetivos intermedios que permitiesen alcanzar más tarde el objetivo principal, aunque sí que tenía algunas reglas para que los personajes abandonasen el objetivo si no era posible conseguirlo.

Sin embargo, en la generación de historias que usen un planificador como motor de la historia (ya sea como Joseph o uno más perfeccionado), la historia puede correr peligro de ir demasiado al grano, o carecer de cierta causalidad. Esto puede ser solucionado introduciendo por ejemplo un sistema de restricciones que hagan que parte del plan necesite de la ejecución de ciertas acciones que de otro modo no se hubieran realizado, como en el sistema de Porteous (Porteous y Cavazza, 2009). Otra manera de solucionar esto podría ser introducir un sistema de valoración dramático, que evalúe la calidad dramática y capacidad de interesar de nuestra historia e influya en las decisiones del planificador. Esto puede ser un problema porque se corre el riesgo de implementar un sistema que esté demasiado ajustada para cierto tipo de historias e invalide nuestra aplicación de generación narrativa para el resto. Sin embargo, realizar una función de evaluación de la calidad argumental de la historia, cosa no trivial, es un recurso ampliamente usado en muchas aplicaciones (y no sólo como solución a un posible planificador narrativo, sino como un completo director del drama) y aunque siempre está sujeto a controversia debido a que es difícil demostrar una métrica objetiva para evaluar historias, funciona razonablemente bien en las que lo usan. Por ejemplo, en *Facade* de Michael Mateas se usa un gestor del drama que regula el ritmo y los sucesos de la historia, al igual que el juego de valve *Left4Dead*, cuyo AI Director controla el ritmo de los enemigos, sucesos, dificultad y música. Una última y novedosa manera de solucionar esto se ve en un estudio llevado a cabo por el IRIS (Cavazza et al., 2009), que utiliza un algoritmo de lógica lineal, que simula relaciones de causalidad y trata los eventos de nuestra historia como proposiciones lógicas.

Además, estas dos estrategias no son únicas sino que hay abundante investigación sobre variaciones o intentos de desarrollo de otras nuevas. Por ejemplo, Monfort además de incluir en su sistema «nn» el concepto de foco narrativo y generar todo un mundo focal sobre un personaje, permitía decidir también el orden narrativo que usaría el narrador. Cavazza (Cavazza et al., 2010) también propone una tercera estrategia centrada en el punto de vista de los personajes. Esta aproximación, permite elegir no sólo en que personaje se centrará el curso de los acontecimientos, sino también elegir que actitud o parecer tomará ese personaje dentro de los permitidos. Así no sólo será distinto contar la misma historia centrándose en dos personajes distintos, sino que podemos tomar a uno como si fuese ingenuo o aventurero, generando historias distintas. Sobre este punto hay que reseñar algo: lo que cambia es no sólo el discurso, sino que cambiaría también lo que se conoce como nivel de fábula según Peinado (Peinado, 2008).

Continuando con la arquitectura de las aplicaciones de generación de historias, aparte de la teoría en la que se inspiren las aplicaciones, cada una utiliza una estructura o algoritmo distinto para aplicarla, siendo por lo general bastante dispares: Minstrel utiliza sentencias que se ejecutan en tiempo real y que transforman y tratan de aproximar una solución al problema que intenta resolver, esto es, la historia que está construyendo de forma creativa; Mexica por otro lado y a pesar de intentar reproducir al igual que la anterior el proceso creativo humano, utiliza una técnica que construye un armazon de historia sobre el que va rellenando la misma, dejando para despues las partes que aun no pueda completar, cosa que hará de forma iterativa; Joseph generaba las historias a partir de reglas de una gramática adaptada a su dominio, mientras que Cavazza usa un potente planificador.

Por último, en cuanto a la evaluación, no suele realizarse un proceso riguroso para la misma, si es que incluso existe alguno, como ya hemos apuntado. Minstrel y Mexica por ejemplo, utilizaban un sistema de encuesta, pero poco riguroso y demasiado susceptible de ser afectado por la subjetividad.

2.3. Creación y evaluación de corpus narrativos

Dado que el objetivo de este trabajo es describir una herramienta de autoría funcional que sea capaz de utilizar un formato intermedio de comunicación entre distintos modelos narrativos, es necesario revisar las herramientas ya existentes y sus principales características.

Una completa herramienta de generación es la aplicación de autoría del grupo de Marc Cavazza (Pizzi y Cavazza, 2008) que surgió a partir de una herramienta de depuración de sus historias de narración interactiva. Dada la necesidad d epoder controlar el flujo de la historial, acabaron desarrollando una aplicación para ello con las especificaciones que adaptasen sus necesidades y que mejorasen las características principales de las herramientas de autoría que existían previamente. Así, en primer lugar, Cavazza distingue dos parámetros cualitativos de las herramientas automáticas para generar historias y que son los que decidieron potenciar: en primer lugar la dimensión de la visibilidad, que indica lo intuitivamente que permite la herramienta hacer posible la comprensión completa de la historia por su usuario, y en segundo lugar la de la generatividad, que describe el grado en que la herramienta nos permite dotar a la historia de comportamientos de alto nivel, sin tener que definir nosotros cada suceso de la historia con detalle.

Así, en valores bajos de visibilidad se encontrarían lenguajes puros de escriptado, en los que es necesario leer código de programación para entender como funcionará la historia, como ocurre en herramientas como ScriptEase, que funcionaba sobre el motor del conocido videojuego NeverWinterNights, y usaba su motor de scripts. En los puntos mas altos, vendrían aplicaciones como INSCAPE, que permiten practicamente ver la historia que se está cre-

ando. Sin embargo, la mayoría de las herramientas parecen alinearse en sólo una de las dimensiones, potenciando ese aspecto en detrimento del otro, cosa que no debería ser así por fuerza, dado que no son aspectos excluyentes. Siguiendo el ejemplo de INSCAPE, en esta herramienta era complicado especificar de forma general sucesos complejos, y había que detallar todo lo que sucedía a bajo nivel. Por tanto, las herramientas con generatividad alta serían aquellas que antes presentaban un perfil bajo en la dimensión de la visibilidad, como Scriptease, que permitía definir con un lenguaje de alto nivel situaciones complejas.

Basandose en estos dos parámetros, el grupo de Cavazza desarrolló una completa aplicación que intentava aunar tanto la posibilidad de ver de un vistazo claramente el contenido de la historia, como la capacidad de poder dirigirla a alto nivel. Analizando esta herramienta, que tiene una gran variedad de controles, ventanas e interfaces, podemos llegar a la conclusión que sus funciones son claramente tres: la modelización del dominio, la generación automática y la ejecución. Con modelar el dominio nos referimos a dotar a la historia de todo el conocimiento que maneje, así como de una estructura básica (estados, objetivos y toda clase de metadatos). Es interesante notar que el modelo usado para el dominio puede condicionar el tipo de historias o sus características de diversas maneras. Por ejemplo, un modelo en el que se tengan en cuenta los sentimientos de los personajes podría generar historias con una tendencia clara a centrarse en los mismos. Otro caso sería un dominio que estructure los objetivos como un plan que generaría historias con una estructura muy clara de subtramas. En general, el modelo condiciona de esta manera las historias porque suele ir asociado también a la manera de crearlas, aunque no tiene por qué ser así. La herramienta de Cavazza llega tan lejos como es posible, permitiendo definir incluso los estados, las acciones, y funciones y las condiciones, que hablando en plata, componen la estructura narrativa de la historia. Esto le permite cubrir prácticamente cualquier historia que deseemos crear dado que sus funciones se crearán ad-hoc según necesitemos pero tiene también una desventaja evidente: el editor humano de la historia debe tener ya en mente toda la estructura narrativa de la misma y saber plasmarla en el modelo. Aunque es discutible, esto puede ser visto como un error, porque evita que un editor que sólo sepa qué historia quiere contar pueda crearla con esta herramienta a menos que pueda extraer de ella las funciones y estados narrativos que necesita, lo que no es trivial ni mucho menos. Por otro lado, como hemos dicho permite modelar cualquier historia que queramos hacer, lo que compensa el defecto. Y dado que la herramienta comenzó como un sistema de depuración del modelo de narración interactiva de Cavazza, los editores iban a conocer siempre el modelo que querían hacer así que cumplía completamente los requisitos. Sin embargo, aunque fuese satisfactorio, el proceso de creación de un dominio es siempre muy susceptible a errores, especialmente si tiene que hacerse uno

nuevo para cada historia y si su tamaño es grande. Por tanto es necesario en estos casos algún sistema de validación de la integridad del dominio, que chequee la consistencia de las funciones, referencias, estados y etiquetas que aparezcan.

La función de generación automática que realiza esta herramienta se nutre del dominio especificado en el paso anterior. En la aplicación de Cavazza la generación es muy completa, permitiendo construir la historia con un plan de golpe o ir paso a paso y permitiendo modificar los derroteros que ésta tomará. Esta función está influida por el método que se use para generar las estructuras narrativas automáticamente, pudiendo basarse en tecnologías de planificación, uso de plantillas, etc. También en esta parte de la aplicación se deberían poder modificar parámetros que influyesen en la generación de la historia. Por ejemplo, en el caso de la generación por punto de vista, debería ser posible cambiar el punto de vista (por tanto cambiando el personaje y la manera con la que vive los acontecimientos), o en un sistema que use restricciones en el planificador, poder modificar las restricciones que queremos que use el mismo.

Por último la ejecución permitiría realizar una visualización de la historia prestando atención a los detalles que se quieran. En esta última función entraría la dimensión del discurso, que indica cómo se cuentan los hechos que ocurren en la historia. Una herramienta ideal, sin embargo, permitiría ver la historia como se desease, para luego generar una sesgada según el discurso escogido, por ejemplo centrada en un usuario, en un rol concreto, o centrada en la trama principal.

Esta estructura es la que podíamos calificar como de estándar a la hora de analizar una aplicación de generación de historias. Todas las que se han revisado tienen una etapa de construcción del dominio, otra para generar los hechos y por último una que nos permita revisar la historia generada. Dentro de estas etapas, se puede profundizar más o menos, considerando que por ejemplo, la herramienta descrita anteriormente se adentra bastante en detalle para cada etapa.

Hemos visto como está concebida la estructura y funcionalidad de este tipo de herramientas, pero es una idea interesante en fijarse en el objetivo o problema que tratan de resolver. Obviamente, una herramienta de generación de historias tiene como meta el generar buenas historias, pero este puede ser un objetivo muy abstracto o abierto. Es cierto que ya hay herramientas comerciales de «construcción de historias» como StoryCraft (Jarvis y Berent, 2010) o StoryWeaver, pero no dejan de ser un apoyo al escritor y tienen un éxito discutible.

Por otro lado, herramientas parecidas a estas que podríamos considerar son algunos editores de niveles de videojuegos, en los que se puede construir la historia interactiva que se jugará a continuación. Si analizamos estos editores dentro de los ejes de visibilidad y generatividad que mencionamos

acerca de la distribución que hacía Cavazza, casi todas estas herramientas tienen valores altos en ambos parámetros. Considerando la visibilidad, antiguamente era complicado deducir la historia a partir del diseño del nivel y del escriptado (ya fuera código puro o una lista de comportamientos más intuitiva) como en los editores de juegos como StarCraft o Heroes of the Might & Magic, sin embargo, ultimamente están proliferando editores en los que puedes «ejecutar» la misma historia que estás diseñando en cualquier momento (lo que se conoce como WYSIWYP, *what you see is what you play*). Sin embargo siguen sin dar posibilidad a saltar a otros puntos de la historia de forma completamente aleatoria. En lo que se refiere a generatividad, dan una amplia gama de opciones al diseñador, ya sea como un escriptado, o mediante listas de condiciones y comportamientos complejos más intuitivos.

Fuera del campo de los videojuegos, podemos ver algunos usos no triviales de estas herramientas. Recientemente, ha surgido la posibilidad de usar la estructura de una historia como sustrato para la anotación de textos. La anotación de corpus es una disciplina que suele usarse como medio para algún otro fin, normalmente relacionado con el lenguaje natural. Podemos poner como usos de estos corpus tres grupos principales de aplicaciones: Extracción semántica de la información como por ejemplo procesar peticiones en un buscador semántico o buscar correferencias, creación de resúmenes, o incluso análisis sobre las tramas de una historia o sus protagonistas y la posibilidad de razonamiento sobre un texto para poder generar conocimiento adicional (que no esté explicitado en la historia. Así, recientemente los investigadores en materia de anotación de lenguaje natural, han visto este gran potencial en la anotación de historias. Chambers (Chambers y Jurafsky, 2010) por ejemplo, propone utilizar un corpus de lo que denomina esquemas narrativos, que son ni más ni menos una estructura formal narrativa anotada, para trabajar sobre ella. La estructura de este corpus la subdivide en dos tipos de relaciones: las de los roles o esquema, que etiquetan cada una de las entidades que aparezcan en la historia con un rol y busca todas sus correferencias, apuntando las acciones en las que toman parte y con qué papel, y la de eventos temporales, que anota entre pares de acciones si una ocurre antes o después de la otra. La inspiración de su trabajo procede del paradigma de script que usó Schank. Sin embargo, los scripts son demasiado costosos de crear (y acumular) y demasiado rígidos para usar en situaciones diversas. Así, en lugar de guiones completos de situaciones, ellos dividen un texto narrativo en acciones atómicas e identifican las entidades que actúan en ellas y con qué papel. Una vez dividido e identificado todo el texto, el resultado es una base de datos que contiene todos los actores etiquetados y las acciones en las que tuvo parte cada uno, además del rol que tuvo en ellas, como puede ser el de sujeto u objeto. Quizás los autores simplifican demasiado en este punto puesto que sólo tienen en cuenta estos dos papeles sintácticos, sin preocuparse de complementos circunstanciales o indirectos. Una vez tienen la

base de datos, calculan por pares la precedencia temporal de un verbo sobre otro. Esta precedencia no requiere de etiquetación externa, pues simplemente se basa en contar el número de veces que ocurre un verbo antes que el otro y viceversa, y si una de ellas es mucho mayor que la otra, esos verbos quedaran marcados como que «ocurren naturalmente en ese orden». Sin embargo a priori podemos ver varios problemas con este procedimiento.

En primer lugar no está decidido que quiere decir «ocurrir mucho más a menudo» y por tanto no parece haber una medida objetiva para decidir si un verbo precede a otro o no. Sin embargo este problema no es demasiado grave, a diferencia del siguiente; en las pruebas que los autores realizaron, se dieron cuenta de que en algunos textos había sucesos que la aplicación designaba como precedentes a otros, cuando en realidad no debería ser así. Por ejemplo, con los términos delito y condena, el texto encontraba mucho más a menudo la palabra condena antes que la otra, con lo que decidía que la sucesión era esa. Los autores explican que esto es así debido a que alguien que ha sido condenado tiene muchas probabilidades de delinquir de nuevo, pero eso sería dotar a la aplicación de una capacidad razonadora que no tiene (o suponemos que no tiene, porque no la han anunciado). Dejando a un lado esa explicación y haciendo un análisis superficial, podemos deducir que dado que el corpus sobre el que se etiquetó eran artículos de sucesos, es lógico que la palabra condena apareciese antes dado que están contados con una narrativa en pasado, contando primero los hechos presentes y después su causa. Podemos sacar dos conclusiones de esto: en primer lugar parece que es importante que esta aplicación no sea automática, puesto que en narrativa, el orden de los sucesos es más semántico que sintáctico, esto es, el tiempo de la acción depende de lo que quiera decir el discurso y no de las reglas de construcción de las oraciones. En otras palabras: La información que necesitamos anotar simplemente no está en el texto, sino en su significado. Por otro lado, realmente es incorrecto decir que un efecto precede a una causa? No es necesario adentrarse en el farragoso mundo de la filosofía analítica, y los autores no hablan de estas relaciones como causales, sino sólo temporales. Puede ser interesante analizar un texto con el estilo narrativo en pasado y establecer todas sus relaciones causales al revés mientras sólo se usen como una relación temporal; en ese caso el tiempo simplemente es una línea que usamos para situar hechos en la historia, sin un significado propio más que el relativo entre cada evento. Sin embargo, en una historia con varias líneas temporales, es absolutamente necesario establecer correctamente tanto la relación temporal como la causal. Por otro lado, esta aplicación nos revela una aplicación interesante: si tomamos como el protagonista de la historia a la entidad que aparece más veces (incluyendo sus correferencias, resueltas al obtener el esquema), podemos trazar una línea de eventos clara con todas las acciones y relaciones en las que tomó parte, haciendo así un resumen de la historia, o incluso, dado que es precipitado establecer el protagonista de

la historia sólo por la frecuencia de aparición, aunque no lo fuese se podría obtener su subtrama. Sin embargo, es posible que se pierda información importante, como algunas causas últimas de los eventos en los que participe. Que sea o no una trama completa es difícil de rebatir puesto que eso depende del estilo de narrativa.

Volviendo a los corpus anotados y siguiendo el camino de la idea de que hay información de la historia que no nos da la sintaxis, Kwong (Kwong, 2010) da una idea interesante sobre otro tipo de información que añadir a dichos corpus: la moraleja o el *quid* de la historia. Al igual que Grosz y Sidner (Grosz y Sidner, 1986), se basan en dividir la estructura de una narración en 3 elementos separados: la estructura del discurso, la atención y las intenciones. La estructura del discurso encaja con lo que vimos anteriormente: las relaciones entre los componentes de la historia y las entidades que participan en ella. El estado de atención del enfoque que tiene cada entidad de la historia, y por último, la estructura intencional describe el propósito de la historia, o incluso el propósito de cada acción de la historia y los aspectos emocionales de la misma. Huelga decir que este elemento es el más difícil de identificar de forma automática, y de ahí que surja la idea de la anotación como solución ideal. Aquí es donde entra el estudio de Kwong: mediante anotaciones, trata de averiguar donde se encuentra la moraleja de la historia, o la información «entre líneas». Además de forma secundaria también intentan identificar el nivel de entretenimiento o interés de cada parte de la historia (o al menos, donde están los eventos que provocan que la historia tenga interés). Realmente su trabajo está pensado como un estudio de viabilidad con aspiración a abrir camino a un método de obtención del *quid* de una historia y los problemas con los que se van a encontrar cuando se desarrolle este sistema. Para ello, realizaron anotaciones sobre un conjunto de historias de Esopo. Las historias eran anotadas por varios anotadores, y cada anotación contenía información tal como la información principal, la que se puede deducir y la moraleja que puedan sacar. También anotaban otros aspectos más comunes vistos en otros sistemas, aunque en este estudio no resulten tan relevantes, como las palabras clave, etc. El primero con el que se encontraron a la hora de anotar información, es que vieron que estaba sujeta completamente a la subjetividad del anotador. Aquí entraría lo que se conoce como acuerdo entre anotadores. Este concepto lidia con el hecho de que a diferente anotador, se generará una estructura distinta de información, puesto que puede no sólo considerar que debe apuntar distinta información cada vez, sino que puede decidir que hay partes de la historia distintas que deben ser anotadas. Otra conclusión que sacan tras analizar los resultados es que a pesar que la historia pueda descomponerse fácilmente en elementos semánticos, oraciones, etc, el significado moral o emocional de la historia (o de la parte en concreto que la tenga) no puede extraerse como un elemento separado, sino que muchas veces aparece de la suma de varios segmentos separados. Esta información

es muy importante, dado que la misma frase, aunque aparezca en dos historias con el mismo significado «semántico», puede permitir que se deduzca un significado moral adicional en una de ellas debido a que otras partes de la misma interactúan con ella.

Por otro lado, ¿este significado moral puede inferirse de alguna manera aplicando razonamiento sobre la información que se puede extraer de cada parte de la historia? Según los autores sí, pero exige una forma de razonamiento intuitiva que por el momento prefieren dejar al anotador. La propuesta preliminar que hacen los autores como conclusión es un sistema de anotación a tres niveles del texto: El nivel estructural, identificaría las funciones que tiene cada fragmento del texto con respecto a la historia. Equivaldría un poco a las funciones proppianas, con ejemplos como: **GOAL**, **ATTEMPT**, **OUTCOME**, etc.

El nivel funcional elaboraría las relaciones narrativas entre las partes de la historia. Los autores enlazan como si fuese un grafo partes del texto, etiquetando las aristas con relaciones tales como circunstancia, contraste, etc. No tienen que ser obligatoriamente relaciones de inferencia lógica, sino que por ejemplo con la relación de contraste se indica un valor emocional o estético del texto, cosa que puede influir en el nivel intencional de la narración: tanto en la moraleja como en el interés.

Por último, el nivel emocional concede valores de polaridad a partes de la historia anotando si son positivos o negativos, y a que entidad se refieren. Es interesante notar que los autores no indican si una etiqueta moral puede aplicarse a un evento o si tiene que calificar sólo a entidades existentes en la historia, aunque lo correcto sería que pudiese. Otra idea notable es que la misma palabra puede tener distinta polaridad en distintos lugares. Como resultado de usar esta estructura de etiquetas por niveles, obtenemos un corpus en el que tenemos una interfaz entre la estructura de la historia y la moraleja. Las posibilidades que tendría esto una vez estuviese en funcionamiento son amplias: medir de alguna forma el nivel de interés de la historia y analizar los puntos en los que surge dicho interés. Detectar los elementos principales y superfluos de una historia, de igual manera que el método de seguir la trama principal del modelo de Chambers, aquí se propone identificar los elementos del núcleo de la historia por ser los que participan en la moraleja y en el interés, aunque esto puede ser discutible: A veces elementos superfluos se añaden para aumentar el interés de los lectores. Sin embargo, es un camino abierto que merece la pena estudiar.

Por otro lado también permitiría analizar las señales lingüísticas y extraer su significado emocional, añadiendo información a un analizador; por ejemplo, en la fábula de la liebre y la tortuga podríamos extraer que la liebre es arrogante con la tortuga y no la respeta. Es más, en idiomas distintos hay elementos sintácticos que añaden esta clase de información, como el chino o el japonés, que indican el nivel del hablante con respecto a una tercera per-

sona. El trabajo de Kwong es un punto de partida interesante y una primera aproximación a la extracción del quid de una historia. ¿Podría hacerse de forma automática? quizá es pronto para que un analizador razone de esta manera con un texto, puesto que como hemos visto, técnicas que hagan esto como la de razonamiento con palabras están sólo en sus inicios teóricos.

Elson y McKeown (Elson y McKeown, 2010) utilizan lo que ellos llaman el grafo de una historia para poder anotar la misma. Este grafo es un modelo de estructura narrativa de la historia, y consideran que es significativo porque cubre todas las factas posibles de un texto y trata a la historia como un todo. Obviamente esta manera de anotación está restringida a textos que puedan considerarse de alguna manera historias. Anotar de esta forma la historia como si fuese una estructura narrativa permite hacer análisis posteriores sobre el género de la misma, los personajes, similitudes con otras historias, relaciones de causalidad implícitas, etc.

En su trabajo presentan Sherezade (Elson y McKeown, 2009), una herramienta de anotación como ellos llaman, pero que también permite generar historias de forma automática. Está pensada originalmente para que a partir de una historia, un anotador extraiga su estructura narrativa y la codifique con el modelo que ellos dan, añadiendo conocimiento extra a la misma. Después será posible hacer operaciones sobre ella, e incluso generar la historia a partir de la estructura y compararla con la original. Esta aplicación sigue las tres etapas de autoría de una historia que vimos antes, aunque ellos dividen a la segunda en 2 etapas y la última, la de generación de la historia, la toman como implícita en la herramienta. Esto es así debido a que para ellos, el objetivo de la herramienta es generar una estructura narrativa anotada de una historia para trabajar sobre ella, y no generar una historia. Por tanto para ellos, se parte de la historia escrita como materia prima. Esto último es interesante, porque al igual que las aplicaciones de autoría pueden servir para generar desde 0 un discurso (la forma «contada» de la historia), tomando la estructura narrativa que hemos construido como un medio para poder generar ese discurso, también pueden verse como herramientas cuyo objetivo es justamente el inverso: extraer y ampliar el conocimiento de un discurso para almacenarlo en forma de un modelo narrativo y luego poder realizar análisis sobre él. Si el proceso de anotación es equivalente al de construir una estructura narrativa merece una discusión aparte dada su naturaleza tan abierta a la filosofía sobre narración. ¿es una historia anotada ya una estructura narrativa o simplemente es una etiquetación? Dejamos abiertas estas preguntas para un posterior trabajo.

Volviendo a la estructura de tres tareas de la aplicación Sherezade (Figura 2.1), la primera es la etapa que llaman de extracción de objetos y es donde identifican el dominio de la historia y todas las entidades que participan en ella. No sólo extraen las etiquetas sino que las dotan de conocimiento añadido. Esto lo hacen buscando en WordNet las entidades, con lo que no

sólo resuelven el problema típico de desambigüación de conceptos (tarea cuya responsabilidad recae en el anotador), sino que añaden un conocimiento ontológico a cada uno que identifiquen, puesto que WordNet dota de una clasificación a las palabras que contiene. La aplicación permite incluir en su dominio tanto conceptos abstractos como concretos, y de entre estos últimos, tanto entidades particulares como generales, como por ejemplo una historia que contenga al perro «Snoopy» y también el concepto de perro en general, ya sea porque aparezcan instancias de perros sin la relevancia suficiente como para ser nombrados, porque se usen para referenciar otros más concretos o como para referirse a ellos como grupo.

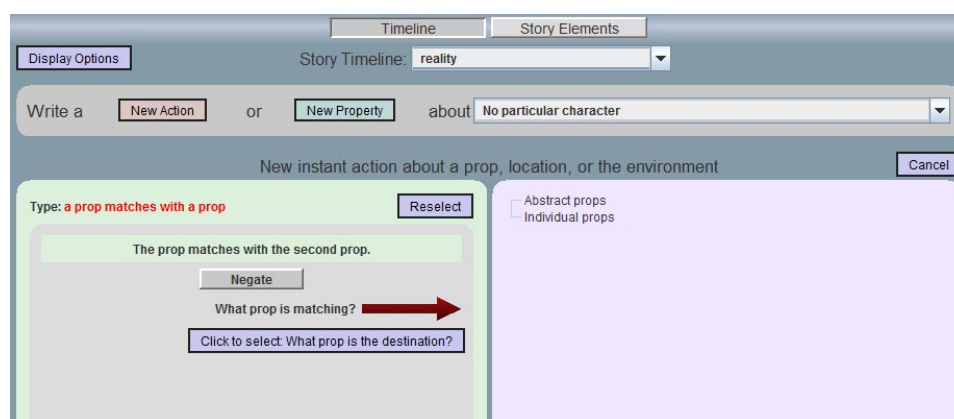


Figura 2.1: Captura de pantalla de la aplicación Sherezade

El paso siguiente de nuevo es el de construcción de los eventos de la historia, que ellos llaman construcción de proposiciones. Para ello han de identificarse todos los predicados de la historia; lo que ellos llaman predicados está presente en muchos otros sistemas y modelos, ya sea en forma de funciones proppianas, relaciones metasemánticas de Schank, etc. Cada vez que se identifique un predicado, se seleccionara la plantilla que use, resolviendo de nuevo así tanto la desambigüación, como el análisis semántico y sintáctico del texto. Las plantillas se seleccionan del corpus de VerbNet en este caso, debido a que contiene una mayor variedad de modificadores para cada predicado que WordNet. Una vez seleccionado el predicado y su plantilla, se seleccionan los sujetos, objetos y complementos que tuviese entre las entidades que se hayan añadido en la primera etapa, y así tendremos construida una lista de funciones básicas de los eventos de la historia. Además dado que estos eventos pretenden ser atómicos, deberán desglosarse las oraciones más complejas y convertirse en una serie de proposiciones, labor responsabilidad del anotador.

La siguiente etapa es la de enlazamiento y tiene como objetivo unir todos los predicados anteriores mediante sus relaciones. Aquí es donde toma forma lo que los autores conciben como grafo de la historia: los nodos son

las proposiciones y las aristas son cualquier tipo de relación entre ellos. Las relaciones que permiten son las temporales y causales clásicas, pero también permiten relaciones modales, que indican el «modo» en el que ocurre una proposición. Este modo permite modelar líneas de sucesos que no ocurren realmente en la historia, como hechos hipotéticos, sueños, etc. Otras relaciones son las que modifican proposiciones (al estilo de la lógica borrosa, pero sin ningún valor de razonamiento interno) y las que unen una proposición con el fragmento del texto original del que se extrajeron. Como vemos, estas dos últimas etapas detallan la construcción de la estructura de la historia y permite modelar cualquiera que queramos. También es suficientemente potente como para indicar acciones instantáneas o sostenidas a lo largo de un periodo. Sin embargo, vemos que la aplicación tiene algunos pequeños problemas a la hora de indicar el orden en el que suceden las acciones en un mismo momento del tiempo, sin embargo esto es más bien el resultado de haber organizado demasiadas acciones para un evento de la historia (aunque la aplicación lo permite). Posiblemente todo esto deriva de lo confuso que resulta el interfaz de usuario y los controles de la herramienta, que hace que etiquetar la historia sea una labor tediosa (además equivale prácticamente a crearla a partir de 0). De todos modos y a pesar de sus problemas es una aplicación muy completa para poder comenzar a generar corpues de historias anotadas.

Otro aspecto muy interesante de esta aplicación es que tiene en cuenta el concepto de acuerdo entre anotadores. A pesar de ser un problema tan obvio, no es tratado en las otras herramientas analizadas, quizá debido a su enfoque orientado a la generación, más que al análisis. En cambio esta aplicación está pensada desde un inicio a la extracción de información de una historia para llenar un corpus e incluso baraja la opción de que una historia pueda ser anotada por personas distintas. De hecho inicialmente hay que indicarle quién va a ser el anotador de la historia, almacenando esta información. Esto permite que hechos pasados por alto por un usuario, puedan estar presentes en la anotación de otro. Los autores contemplan una métrica de evaluación para la medida de acuerdo entre dos anotaciones de la misma historia, basada en la métrica para comparar proposiciones de WordNet de Lin (Lin, 1998). Usando esta métrica se puede designar un valor para la coincidencia de la información extraída por cada anotador y comprobar su similaridad. Los resultados que ofrecen no son malos a la hora de extracción de información, sin embargo, no dan un método para unir de alguna manera la información disjunta de ambas anotaciones. La aplicación de momento carece de soporte para modelar tanto la «moraleja» final de la historia como los valores morales de las acciones, y tampoco permite un sistema de creencias deseos y objetivos para los personajes.

Por lo general las aplicaciones más complejas, como la de Cavazza, permiten hacer esto hasta cierto punto, por ejemplo para indicar si se quiere

contar la historia desde un punto de vista de algún personaje con alguna ética concreta. A este conjunto de características de la historia los autores de Sherezade la denominan capa interpretativa, que viene a ser similar a la capa de fábula de Peinado. Toda esta convergencia hacia el mismo paradigma es una muestra del problema del contexto mencionado antes en el que la investigación parte de cero con su propio modelo, sin trabajar sobre ciencia ya asentada, produciendo al final resultados y conclusiones similares.

Como hemos visto, multitud de herramientas de edición narrativa han ido surgiendo en los últimos años, con muchos problemas y objetivos comunes. En el siguiente capítulo presentaremos nuestra propuesta de modelo para editar historias narrativas y mostraremos las novedades que éste supone.

Capítulo 3

Propuesta: Corpus narrativos y colaboración masiva

RESUMEN: En este capítulo presentamos nuestro modelo para la edición de historias narrativas, sus objetivos y sus ventajas. Comenzaremos por elaborar una lista de los requisitos que nos hemos propuesto cumplir, para a continuación describir nuestro modelo genérico y el dominio al que puede enfocarse. Por último hacemos un hincapie especial en el uso de técnicas colaborativas en la web, como herramienta para conseguir distintas ventajas que serán convenientes para nuestro sistema.

3.1. Requisitos del sistema

El propósito de este trabajo es presentar un sistema de creación y edición de corpus narrativos que permita obtener muchas historias dentro de un dominio concreto pero que al mismo tiempo sea sencillo e intuitivo de usar. Además nuestro modelo permitirá la colaboración de múltiples usuarios, usando las técnicas de computación colaborativa aplicadas a la web 2.0. Así pues, debemos comenzar explicando por qué nuestro objetivo es el de un sistema de edición de historias narrativas. Desde el principio, en el campo de la narración digital ha existido un cuello de botella que limita el desarrollo, la utilidad y el uso de las herramientas que va desarrollando el panorama investigador. Concretamente, es el proceso de crear historias el que lastra el resto de disciplinas, puesto que al no disponer ni de autores disponibles o de una base de historias creadas y anotadas, los sistemas de generación de historias no pueden demostrar completamente su potencial.

En un estudio sobre herramientas de creación para narración interactiva, Medler y Magerko (Medler y Magerko, 2006) proponen una serie de condiciones para una herramienta de este tipo. Si bien las historias que crearemos

en el presente estudio no son interactivas, sí que podemos fijarnos en la lista y centrarnos en la parte «clásica» de la generación de historias, que también es aplicable. Los principios que establecen son seis y son los siguientes:

- La edición debe ser independiente tanto del soporte de la historia como del contenido.
- Debe ser capaz de valorar de alguna forma la historia. Esto incluye comprobar errores de coherencia o pobreza narrativa
- Debe ser usable, lo que se aplica tanto a interfaz como al sistema de edición, que deben ser sencillos pero eficientes.
- Debe permitir representar completamente el dominio de la historia.
- Debe permitir configurar el ritmo y tiempos de la historia.

Como vemos, todos estos puntos son relevantes de alguna forma aunque nuestra herramienta no genere narración interactiva así que utilizaremos estos principios como requisitos iniciales a grandes rasgos. Así que nuestro sistema deberá ser lo suficientemente completo como para crear historias realistas y completas, pero al mismo tiempo debe permitir crearlas en el menor número de pasos posible y de forma sencilla. Pero, ¿a qué nos referimos con una historia «realista»? Está claro que si el dominio de historias sobre el que nos movemos es de ficción, no podrá tratar sobre sucesos reales, sino que a lo que nos referimos es a tener como objetivo producir historias con el argumento suficientemente complejo como para que sean consideradas historias completas, con todas las de la ley. Sin embargo, es muy difícil establecer una medida de qué se considera realista y completo y qué no. Por tanto, la única manera que tenemos de evaluarlo será comparar con historias de la misma temática, pero que hayan sido producidas por un autor humano y aceptadas por los medios de comunicación (películas, novelas, videojuegos, etc.). Así pues, lo que queremos conseguir es que el sistema propuesto nos permita construir historias análogas a las ya existentes, en complejidad y argumento.

Por otra parte, queremos que esta herramienta, no sólo nos permita crear una historia, sino que, viendo las posibilidades que nos ofrece internet, y en particular la computación colaborativa, queremos permitir que los usuarios puedan completar, mejorar o editar otras historias, permitiendo varios anotadores o editores para la misma historia. En la presente investigación se pretende analizar si permitir esta clase de colaboración es beneficioso y aporta valor añadido a las historias creadas y anotadas.

Acerca de la usabilidad nos enfrentamos a un problema a la hora de elegir donde mantener el balance entre ofrecer un modelo muy completo y que el modelo sea permisivo con los usuarios permitiéndoles que no tengan que bajar a una granularidad muy fina a la hora de descomponer la historia que

tienen en mente y construirla con él. Además de la complejidad o completitud del modelo, otro asunto a tener en cuenta es la usabilidad de la interfaz. Como ya establecía Cavazza, es importante que mirando a simple vista una historia modelada, un usuario fuese capaz de hacerse una idea de la historia en sí, y de ahí la creación del parámetro de la visibilidad como métrica para evaluar un editor de historias. En el caso de conseguir nuestros propósitos, dispondríamos de una herramienta práctica y usable, colgada en la web, y que permitiese crear cualquier historia que el usuario desee, y también ver o editar las ya creadas.

3.2. Modelo de estructura narrativa

El modelo que proponemos pretende mantener un equilibrio entre la facilidad de uso y la alta visibilidad (tal y como la define Cavazza) y la potencia. Así, pretendemos que los usuarios sean capaces de crear, editar y anotar historias sin formación previa o conocimientos sobre teoría narrativa u ontologías, y que además viendo el resultado de una historia ya creada, se puede uno hacer una idea clara y completa de toda la historia. Además, en cuanto a la potencia del modelo y su completitud, las historias creadas deberían poder ser todo lo complejas que se quiera mientras lo permita nuestro dominio. Sin embargo, no bastaba con desarrollar un modelo completo, sino que además debía tener una granularidad lo bastante grande como para permitir acciones generales sin tener que especificar muchas acciones más elementales e irrelevantes para la historia, pero que a su vez fueran acciones lo suficientemente específicas como para cubrir todos los tipos de historias que permita nuestro dominio con exactitud. Para esto, el requisito que tuvimos en mente siempre al diseñar los eventos que formarían nuestro modelo fue que cada evento pudiese representar un suceso claro en la mente del usuario, tanto si estaba editándolo, y por tanto podía plasmarlo directamente de sus ideas, como si está viendo la historia construida. Es difícil definir cómo debe ser un «significado semántico» o que límites tiene a la hora de representar un suceso. ¿Debe ser un verbo o plantilla como las de Schank? ¿Un gránulo semántico como en el procesamiento con palabras de Zadeh? Lo que hemos visto que tienen en común todas estas teorías es que la elección de cada unidad de expresión (por llamarla de alguna manera) es completamente ad hoc al dominio que tengamos y que aunque esto nos permite ajustar nuestro modelo de forma perfecta al dominio, acarrea una gran complicación a la hora de desarrollar el motor de razonamiento que lo procese. En el sistema que proponemos no hay un motor de procesamiento, aunque sí un control de restricciones, no tenemos esta desventaja, por lo que el punto de partida que tomaremos será el mismo que el de las dos teorías dichas (y el de muchas más). Por otra parte, aunque elijamos los eventos ad hoc para nuestro dominio de historias, tenemos que mantener un equilibrio entre eventos muy

generales, que permitan múltiples interpretaciones y que al mismo tiempo sea lo suficientemente específicos para que sólo cubran el significado semántico de ese evento. Por ejemplo, un evento de Hallazgo (en el capítulo siguiente se verá más en detalle un modelo completo de ejemplo que usamos y cada uno de sus eventos) debe representar cualquier tipo de hallazgo y en cualquier situación como pueda representar el usuario, pero no podrá representar otra cosa que no permita el concepto intuitivo del evento, esto es, un «hallazgo». Así, en resumen, el usuario podría utilizar los eventos, que de por sí son específicos para construir situaciones prototípicas en las historias de nuestro dominio, de la manera mas flexible posible. Para hacer esto posible, es importante precisamente concretar y cerrar bien el dominio sobre el que vamos a trabajar (o más bien nuestro editor) porque así nos permitirá definir el alcance y tipo de los eventos y el número y clases de elementos que tendrá nuestra historia. Este será el primer paso para construir nuestro modelo: estudiar y formalizar un dominio. Para ello, debemos definir un dominio en el que sea posible:

- Delimitar el número de eventos posibles.
- Delimitar las clases de elementos participantes en las historias.
- Establecer a priori las compatibilidades entre elementos y eventos.

Con esto queremos decir que en nuestro dominio sólo serán posibles determinados eventos y elementos, y que además cada uno de estos últimos tendrá un uso muy concreto, mientras que lo verdaderamente flexible es el evento. Para poder extraer esta información, una vez decidido el dominio de nuestra aplicación, decidimos analizar exhaustivamente una gran cantidad de historias pertenecientes al mismo y leer los estudios sobre ellas para poder construir un modelo sólido que abarcara dicho dominio. Incluso aunque no haya demasiada literatura académica sobre dicho dominio si que habrá intentos de análisis serios generalmente. Al hacer esto, podremos hacernos una idea clara de los eventos que tendrá nuestro modelo para ese dominio, y podremos formalizarlos. Para ello tendremos que realizar otro paso previo, que es extraer, de nuevo del análisis que hemos hecho del dominio, todos los tipos de elementos de esas historias y sus usos en los eventos. Debemos clasificar los Elementos en tipos y establecer alguna clase de propiedades en ellos, como por ejemplo, decidir que necesitamos un tipo de elemento llamado Personaje, cuyo tipo es «Niño» y «Adulto». El tipo no debe ser meramente cosmético, sino que debe influir en cómo se comportará este elemento en relación con los demás y con los eventos. Además de eso, habrá que decidir si los elementos son únicos o instanciables, permitiendo en el segundo caso asignarse todas las veces que se quiera a un evento, mientras que en el primero como máximo sólo podrá aparecer una vez en cada uno. Una vez hecho esto, deberemos especificar, para cada uno de los eventos que hemos aceptado como nuestros,

los tipos de elemento que se le pueden asignar, y qué significa cada uno o qué papel juega. Como dijimos, nuestros eventos son semánticos, con lo que una vez instanciado, no hay que indicar que propósito o uso tiene un elemento asignado a un evento, sino que de eso se encarga el editor y el anotador. Finalmente, una vez tengamos completamente especificado nuestro modelo, debemos tener en cuenta que cuando lo usemos para construir historias narrativas, el resultado será una lista ordenada de eventos con ciertos objetos asignados a ellos, y de forma secundaria, también tendremos un conjunto con los elementos que hemos usado.

Como vemos, esta extracción de requisitos y formalización del modelo es no sólo *ad hoc* al dominio que hayamos decidido, sino que en principio habrá que hacerla manualmente. Sin embargo, lo que realmente importa es que como resultado habremos obtenido una instanciación usable de nuestro modelo, que permite la flexibilidad de poder realizar cualquier historia dentro de nuestro dominio de forma sencilla. Por supuesto existirán dominios que no se puedan adaptar al modelo, debido a que sean demasiado amplios o generales, o que permitan hacer cosas contrarias a nuestro paradigma de eventos con significado semántico, pero si se acotan mejor, entonces si se podrá aplicar nuestro modelo.

Una vez diseñado el modelo concreto aplicado al dominio elegido, podemos pasar a construir las historias pertenecientes al mismo. Para ello, simplemente necesitaremos definir los Elementos que tomarán parte en la narrativa, y sus propiedades, y luego ir creando los Eventos en el orden en el que suceden en la historia. A los Eventos les añadiremos todos los Elementos que participen en ello, siempre cumpliendo las restricciones que establecimos en nuestro modelo. Además, a los eventos y objetos se les podrá añadir texto y anotaciones, para añadir información no explícita, texto personalizado, aclaración de las causas de ese evento, etc. Una vez completados todos los Eventos que ocurren en la historia, habremos terminado. Como se ve, nuestro modelo en principio sólo permite una línea temporal de eventos ordenada, aunque si construimos bien el modelo, podemos permitir eventos que cambien de línea temporal y describan saltos de distinto tipo. Esto es debido a que nuestra lista de eventos sólo representa realmente el orden de los eventos en el discurso, y no tienen por qué ser necesariamente el orden de los eventos en la historia. La razón de mantener sólo una línea aparente es la de maximizar la visibilidad de nuestra historia, permitiendo así en todo momento saber en qué orden se cuentan al usuario los eventos. Además, esto también hace que nuestro modelo pueda guardarse y cargarse en un formato relativamente sencillo, computacionalmente hablando.

En resumen, podemos ver que el paso de construir historias a partir del modelo terminado, es bastante sencillo y rápido, puesto que pretende ser un modelo puramente intuitivo. El paso complicado como decíamos antes, es el de especificar el modelo y elegir cuidadosamente el alcance de nues-

tro dominio, pues esto es lo que permite que los Eventos, los componentes atómicos de nuestras historias, sean intuitivos al tiempo que se intenta que sean conceptos formales y específicos a la hora de construir la historia.

3.3. Crear y evaluar corpus mediante colaboración masiva

Volviendo a la propuesta de una herramienta web, ¿por qué decidimos hacerla en internet? La respuesta es sencilla: en los últimos años, la tendencia actual es la de migrar todas las aplicaciones a la web, siguiendo ejemplos como el de Google con sus googleapps o el de Microsoft y su apuesta por la computación en la nube, es deseable tener nuestras aplicaciones en Internet, no sólo porque permiten descentralizar parte del proceso (cuando no todo), sino porque permiten su difusión al gran público de una manera mucho más inmediata y efectiva. En el caso que nos ocupa, nos interesa especialmente mantener en la red nuestro sistema de edición de textos narrativos porque nos va a permitir conseguir un logro que ya buscaban aplicaciones de anotación previa: la compatibilidad de varios editores y la validación de anotaciones y textos. No hay más que ver el sistema de edición de páginas como la wikipedia, que permiten ser editadas o corregidas por cualquiera. Sí, esto también abre el riesgo a vandalizar contenidos o a introducir errores, pero la teoría de la computación colaborativa nos dice que es más probable que el resultado (sobre todo a largo plazo) sea correcto. Si bien es posible que no sea la mejor herramienta de edición narrativa, sí que podemos decir que es la primera que se hará siguiendo esta filosofía, y posiblemente este será el camino que se siga en el futuro.

La colaboración masiva es un concepto propuesto por primera vez por Sir Francis Galton, en 1907 (aunque no con ese nombre en concreto, sino que escribió un artículo sobre el fenómeno que describe) (Galton, 1907). Galton se fijó en un concurso de una feria agrícola que se trataba de adivinar el peso de una res una vez hecha su matanza y deshuesada. Galton pensó inicialmente que la mayoría de la gente no tendría mucha experiencia sobre la constitución de animales de granja y que sólo unos pocos expertos, como carniceros, veterinarios o ganaderos podrían acercarse al peso real, por tanto, la media de las respuestas estaría muy desviada de la correcta. Sin embargo, al final del concurso vio con sorpresa que el promedio de las respuestas (realmente usó la mediana, no la media) se acercaba mucho al peso real, y decidió investigarlo, llegando a la conclusión de que el resultado de la decisión efectuada por la multitud suele ser mejor que la que haga un solo individual, y no simplemente igual a la del más informado o sabio de sus miembros.

Este efecto era difícil de poner en práctica fuera del terreno de la estadística, sobre todo para obtener resultados rentables o útiles, así que hasta la introducción de internet no se avanzó mucho en el tema. Así, muchos después,

James Surowiecki publica en 2004 el libro *The Wisdom of Crowds* en el que analiza este fenómeno (Surowiecki, 2004). Según él, la toma de decisiones distribuida en grupos grandes tiene ventajas como velocidad, confiabilidad y mayor resistencia a la manipulación (mucho más que las hechas por pequeños grupos de expertos). Sin embargo, establece una serie de reglas para poder tener una multitud sabia (lo que denomina *wise crowd*), que pueda tomar decisiones y por tanto tener todas las ventajas descritas antes:

- Cada persona individual del grupo debe tener su propia información privada, aunque ésta esté distorsionada. Esto favorece que haya diversidad de opiniones en el grupo.
- La opinión de cada persona no debe estar influenciada por la de las demás. Esta condición como vemos es extremadamente complicada de cumplir según el caso en el que queramos aplicar la colaboración masiva. debido a que muchas veces es necesario conocer los resultados y opiniones de otros miembros del grupo, lo que influye irremisiblemente en la opinión de los que las conocen. Sin embargo, existen formas de evitar esta «contaminación» de opiniones.
- El conocimiento de los miembros del grupo debe ser lo más diverso posible. Cada persona suele tener más conocimiento en su especialidad, y es deseable que estas sean muy variadas.
- Debe existir una forma de sondear cada respuesta u opinión y obtener una conclusión colectiva. Dependiendo del problema que estemos tratando esta labor puede ser muy complicada.

Las aplicaciones de la colaboración masiva, cubriendo campos como el mercado de valores, la predicción empresarial o electoral e incluso la seguridad antiterrorista. Sin embargo, por lo que respecta al método que propone este trabajo nos vamos a centrar en la más puramente computacional y la relacionada con la Web 2.0. El término Web 2.0, acuñado por O'Reilly en una conferencia en 2004 (O'Reilly, 2005), es una concepción nueva del uso de la web en el que prima más el servicio y la actividad de los usuarios que el contenido, puesto que éste es generado por los mismos. Hemos de decir que el concepto de la colaboración a la hora de crear software no es nueva, pues es parte de la filosofía del código abierto. Lo que sí puede considerarse nuevo, es el estudio de si esta clase de colaboración es exitosa cuando se aplica a los contenidos. El mejor ejemplo que podemos encontrarnos es el de wikipedia, aunque también es el más controvertido. Fuera de él, tenemos otros como por ejemplo, el concurso que publicó Goldcorp para encontrar depósitos de oro en Ontario, colocando disponibles al público todos sus datos geológicos y consiguiendo la solución gracias a un grupo australiano.

Por otro lado, esta teoría tiene muchos críticos, que esgrimen que no siempre se consiguen mejores resultados cuando se trabaja en multitud, citando ejemplos como el de Kasparov contra el mundo en 1999 (Kasparov y King, 2000), en el que se jugó una partida de ajedrez del campeón ruso contra internet, concretamente el portal de MSN Gaming Zone, que organizó la partida. Las jugadas del jugador «mundial» eran discutidas y decididas en foros y a través de votación, con 24 horas para pensar los movimientos cada uno. La partida la ganó Kasparov, definiéndola éste como la partida mas complicada y que mayor esfuerzo le había costado ganar. Se podría decir que si la teoría de la sabiduría de la smasas es correcta, no debería haberla ganado, pero esto no es del todo cierto: Kasparov tambien participaba (al menos leyendo) en las discusiones de movimientos que hacían los jugadores de internet. Si esto fue una ventaja decisiva o no ya es objeto de estudio.

Surowiecki cita cinco situaciones en las que la computación de masas no funciona, y la decisión o solución encontrada recibe un desvío. Las situaciones son la siguientes:

- Si los individuos de la multitud son demasiado homogéneos, el conjunto de respuestas obtenidas tendrá muy poca variedad.
- Si ciertos componentes de la multitud sólo pueden acceder a partes de la información disponible, la respuesta obtenida será sesgada.
- Si el conocimiento de parte de la multitud es considerado a priori mejor que el de otra de las partes es posible que se pasen por alto datos vitales.
- Cuando las elecciones de cada individuo son visibles se corre peligro de caer en una cascada de información, en la que una vez que las decisiones iniciales tomadas por algunos individuos son suficientemente informativas, las subsiguientes, tomadas por el resto simplemente siguen el patron dictado por los primeros.
- Cuando la decision puede ser influenciada o puede afectar a factores emocionales, los individuos pueden padecer de presión social o comportamientos gregarios.

En general, y dependiendo del problema a resolver y del método utilizado, puede ocurrir que ciertos de estos problemas sean triviales de resolver, o dificultades endémicas. El caso de la fiabilidad de Wikipedia, mencionado antes, adolece del último de los problemas principalmente, mientras que otros son casi inexistentes.

Por otro lado, ya existe un precedente en la narratología computacional que intenta usar el poder de la smultitudes en su favor y además para obtener un corpus narrativo. Say Anything (Swanson y Gordon, 2008), surgida en 2008, es una aplicación de narración interactiva que se nutre de los textos

de un corpus de blogs escogidos de internet para crear historias. La aplicación permitía al usuario interactuar con ella y guiar la narración, dejándole que inicialmente escribiese un fragmento de la historia que éste quería hacer, para a continuación, generar automáticamente el siguiente fragmento de la historia, pasándole de nuevo el turno al usuario y así hasta que éste considerase que la historia había terminado. Para obtener las respuestas, Say Anything comparaba la entrada del usuario con su base de datos de frases de todos los artículos de los blogs que almacenaba usando un motor de ocmparaciones, que devolvía el fragmento que considerase que mejor se adaptaba al del usuario, usando métricas como índices de valores para las palabras. Una vez encontrada la frase o fragmento que más se parecía según el comparador, el sistema devolvía la siguiente del mismo documento. Esta estrategia probó resultar muy útil a la hora de dar interactividad, pero tenía una serie de defectos que hacían que las historias resultantes no fuesen demasiado coherentes. Estos defectos venían más por el lado del análisis de lenguaje natural (que no existía más que por palabras sueltas): no tenía en cuenta las correferencias, con lo que en una frase no usaría las palabras referenciadas para buscar comparaciones, usando por ejemplo los pronombres en lugar del nombre al que se refiriese. Tampoco recordaba el tema o los datos aparecidos anteriormente, con lo que el tema de la historia podía variar completamente de una frase a otra, o presentar hechos incoherentes sobre el estado del mundo de la historia, como por ejemplo no recordar que se ha dicho que era verano y luego responder con una frase sobre el invierno. Sin embargo, Say Anything presenta una aplicación novedosa en el campo de la narración, abriendo un importante camino, y mostrando que internet y el uso de corpus colaborativos podría ser un enorme beneficio.

Para conseguir una computación colaborativa fiable, en el presente sistema se han intentado cumplir las cuatro condiciones necesarias que establecía The Wisdom of Crowds, siendo la más difícil de conseguir la de la independencia de decisiones entre individuos, dado que debíamos permitir a todos los usuarios ver y modificar las historias de los demás. Sin embargo, en este caso no necesitamos evitar en demasiada medida que los resultados de un usuario afecten a otro. En concreto, existía el dilema de si permitir ver a los demás usuarios los comentarios, puntuación y anotaciones de las historias. Tomamos la decisión de permitirlo debido a que si no, era posible que no se transmitiese toda la información de la historia a los usuarios. Puede ser más adelante objeto de otro estudio el averiguar si es mejor permitir a los usuarios ver las anotaciones de los demás, pero mantener la historia y anotaciones originales completamente públicas. Una vez conseguido un editor que sea capaz de gestionar la colaboración de usuarios, podremos ver si esta característica nos ayuda con el anotado de historias mediante varios usuarios a la hora de generar un corpus abundante de historias. En el apartado de análisis de los resultados, estudiaremos más a fondo el efecto que tiene el

poder modificar, puntuar o ampliar las historias de otros sobre el resultado.

Como conclusión podemos ver que la principal razón de hacer la herramienta prototipo de nuestra propuesta en forma de aplicación web, es la de poder permitir la colaboración masiva para crear y evaluar las historias obtenidas mediante nuestro editor. Por otro lado, otra ventaja que obtenemos de implementar nuestro sistema con tecnologías web, es que nos garantiza (al menos teóricamente) que la interfaz tiene ya un paradigma aceptado y nos provee de una capa intermedia de herramientas para construirla, como pueden ser los framework de jquery, mono, yahoo o incluso mediante javascript nativo.

Capítulo 4

Caso de estudio: Zombie Story Maker

RESUMEN: En este capítulo mostraremos el prototipo de ejemplo que hemos desarrollado para poner en práctica nuestro modelo narrativo. Inicialmente describiremos el alcance que tiene nuestro dominio escogido y sus características. Después, analizaremos a fondo el conjunto de las historias narrativas que contiene dicho dominio y sus estructuras, explicando como las hemos convertido a nuestro modelo y mostrando ejemplos, para finalmente repasar en detalle la aplicación de ejemplo creada y explicar su modo de uso.

4.1. Dominio de la aplicación

El dominio escogido para desarrollar la aplicación que ilustrará las ideas del capítulo anterior es el de las películas de terror, concretamente las historias sobre muertos vivientes, también conocidas como pertenecientes al subgénero zombi.

A la hora de escoger una aplicación que se ajustase a ejemplificar las conclusiones que hemos sacado y también a demostrar que podíamos aplicar las teorías de desarrollo de historias a la computación cooperativa, hemos decidido hacer una aplicación ligera en web de edición de historias de zombis

¿Por qué hemos escogido el género de historias de zombis? La respuesta obvia es para limitar el dominio, pero, entonces, podríamos habernos conformado con historias de terror, sin tener que especificar tanto el género. Sin embargo, el subgénero zombi en concreto, tiene una serie de aspectos que por su uniformidad y repetitividad hacen de las historias de zombis un subconjunto mucho más abordable, y puesto que la intención de este trabajo es realizar un corpus narrativo mediante editores que permitan computación

cooperativa, nos basta con el menor subconjunto de tipos de historias completas que podamos obtener. En el análisis que mostraremos a continuación, se explicará por qué es posible hacer historias de zombis «de libro», o utilizar patrones narrativos fijos.

El concepto de zombi (un humano muerto que vuelve a la vida pero sin retener su inteligencia y personalidad) ya era popular desde la época del romanticismo, apareciendo seres similares a estos en las historias de Mary Shelley y Edgar Allan Poe e inspirando a autores posteriores como H.P. Lovecraft (Lovecraft, 1973). El imaginario de estas criaturas venía de viejas leyendas medievales, pudiendo trazarse incluso hasta su aparición en un relato de las mil y una noches. Sin embargo, todas estas historias no podían aún ser consideradas como historias de zombis modernas, puesto que estas criaturas tenían características a menudo heterogeneas y simplemente eran más monstruos resucitados o apariciones. El germen de este género comienza a gestarse a partir de la década de los 50, cuando a raíz de la novela «Soy leyenda», de Richard Matheson, que introducía el concepto una plaga a gran escala, y lo mezclaba con el mito del vampiro, dado que los efectos de la plaga provocaban todas las características de los vampiros clásicos (intentando explicarlas pseudocientíficamente). Aunque aun faltaban algunos conceptos que definiesen al zombi «moderno», como la capacidad de infectar a otros (aunque esto es discutible, ver más adelante) y obviando el hecho de que las criaturas eran más bien vampiros, esta novela inspiró después varias películas y otras historias. Unos años después aparece la primera historia moderna de zombis: «La noche de los muertos vivientes» de George A. Romero (Harp-er, 2005). Como nota curiosa, anterior a esta película, es la publicación del comic «Los pitufos negros» que es una historia completa de zombis al estilo moderno con todos sus elementos (pero con pitufos, claro).

Los zombis que mostraba Romero fueron la base de todas las historias modernas, incluido el concepto de apocalipsis zombi, en el que la sociedad moderna se derrumba (usualmente desde dentro) ante el imparable número de los mismos. En este trabajo pasaremos de puntillas sobre las alegorías y metáforas que quiso darle Romero (y los autores posteriores) al significado de estas historias: usualmente son un reflejo del peligro del consumismo, la despersonalización de la sociedad y la ineptitud de los gobiernos; esto sería objeto de debate en otros foros y a pesar del interés que pueda suscitar, no es el objeto de este trabajo.

Posteriormente a la primera película de Romero, surgieron muchas más películas de la misma temática, siguiendo el mismo patrón, introduciéndose novedades significativas cuando O'Bannon comenzó su saga de películas y creó sus propias reglas para los zombis. Novedades como la apetencia de comer cerebros, capacidad de habla y memoria y de reanimar cualquier parte del cuerpo (todas estas en la saga de O'Bannon, aunque después no han sido demasiado utilizadas). También se enriqueció y fortaleció el concepto

del origen del zombi a causa de una enfermedad/química artificial, cuyos comienzos pueden trazarse hasta Lovecraft con «Herbert West–Reanimator» (si obviamos al Frankenstein de Mary Shelley, que técnicamente es un zombi) popularizando el origen de la enfermedad en un laboratorio científico o militar (Lovecraft et al., 1999 (1922)). O'Bannon incluso añadía en sus películas la capacidad de contagiarse por vía aérea, dado que el compuesto que creaba a los zombis era un gas: la trioxina, inspirándose en una película 4 años anterior dirigida por Bruno Mattei, «Virus». Fue el mismo O'Bannon (junto con John Russo, que ejerció de director) el que se separó de la línea marcada por Romero (Russo, 1978) introduciendo las dos grandes diferencias de estilo que pueden apreciarse entre las historias de zombis:

- El sistema de convertirse en zombi: Las historias iniciales de zombis simplemente mostraban algún tipo de manipulación del cuerpo muerto como requerimiento para reanimarlo, y posteriormente surgió el concepto de plaga o enfermedad, lo que introducía el contagio de una u otra forma. En la saga de Romero, que es la que inició el género, la gente simplemente vuelve como zombi poco después de morir (aunque obviamente si una persona muere al ser devorada por un zombi o por las heridas que estos la infligieron, volvería como uno más), pero O'Bannon y otros introdujeron el concepto de contagio mediante el propio zombi (a través de mordiscos, garras, etc.), convirtiendo cualquier infección en mortal, aunque morir de otro modo no reanimaría un cadáver. Esto produjo una de las dos clasificaciones importantes en las que se pueden incluir los zombis: necesidad o no de una infección para volver como zombi.
- La velocidad y maniobrabilidad de los zombis: Inicialmente los zombis en ficción habían sido criaturas lentas y torpes, cuyo poder resistía en su mero número y capacidad de conversión, pero por lo demás no presentaban más amenaza. Rompiendo con esta idea tradicional, surgieron historias en las que los zombis podían correr saltar e incluso en algunas historias, a manejar herramientas. Finalmente quedó en la tradición el concepto de «zombis que corren», en contraste con los tambaleantes.

Como vemos, las historias modernas de zombis tienen un origen relativamente reciente, y podemos establecer una serie de patrones comunes a todas ellas, que nos facilitan mucho la creación o modelización de nuevas historias:

- Esencialmente sólo habrá dos conceptos de personaje: el humano y el zombi. A partir de aquí, los personajes humanos podrían tener cualquier arquetipo típico pero no es relevante para los sucesos de la historia; independientemente de sus motivos, personalidades o conflictos, sólo es importante saber que los humanos son los que corren peligro y además se pueden convertir en zombis, mientras que los zombis solo tienen un objetivo (comerse a los humanos)

- Los escenarios en los que sucede la historia son meramente cosméticos, pues prácticamente sólo importa saber si hay o no zombis en ellos. En estas historias el escenario se ve reducido a un mero decorado. Incluso a pesar de ser historias de terror, no es necesario que el escenario sea el típico de estas historias. Por lo general, lo máximo que tendrá de relevante un lugar de la historia será que contenga cierto objeto en particular.
- Los objetos sólo tienen 3 utilidades generales: actuar como arma, permitir de alguna forma el acceso a un escenario o ayudar a bloquear el acceso al mismo. En estas historias también hay objetos que den información (herramientas de trama), pero realmente la historia puede suceder igual sin éstos.
- Los sucesos o eventos que ocurren en la historia pertenecen a un subconjunto acotado: en general las películas y novelas de zombis suelen ser muy parecidas entre sí debido a esto.

Usaremos estas características canónicas para construir nuestro modelo, que explicamos en el siguiente apartado.

4.2. Estructuras narrativas del subgénero zombi

En esta sección presentaremos el modelo narrativo escogido para estas historias, que hemos desarrollado a base de extraer los elementos comunes de gran parte de las historias de zombis (en cine, libro, videojuegos, televisión, etc), y mostraremos el respaldo que tiene con ejemplos. El modelo narrativo propuesto divide la historia en dos conjuntos de entidades: los eventos, y los elementos. Siendo los eventos las cosas que ocurren en la historia, ordenadas en el tiempo, y los elementos los participantes que toman parte en ella (ya sean animados o inanimados). Eventos: Como decimos, hemos definido el elemento unitario de la historia como el Evento. Un Evento es un suceso que puede ser más o menos complejo y que en sí mismo ya puede contar una historia. De hecho es posible realizar una historia con sólo sucesos, e incluso con un sólo suceso. A la hora de definir la lista de Eventos disponibles, hemos querido huir de acciones narrativas demasiado simples, dado que no estamos componiendo un planificador (como establece Marc Cavazza, sería un error obligar al encargado de la creación de la historia a que tuviese que colocar 20 eventos en orden para definir una acción que tiene por ella sola nada más que un significado narrativo.

Otra razón para mantener los eventos narrativos a un nivel lo más alto posible era permitir al usuario que entendiese a primera vista para qué podía utilizar cada uno, siendo lo más intuitivos posible, pero la mismo tiempo permitiendo crear historias lo suficientemente distintas y completas mediante

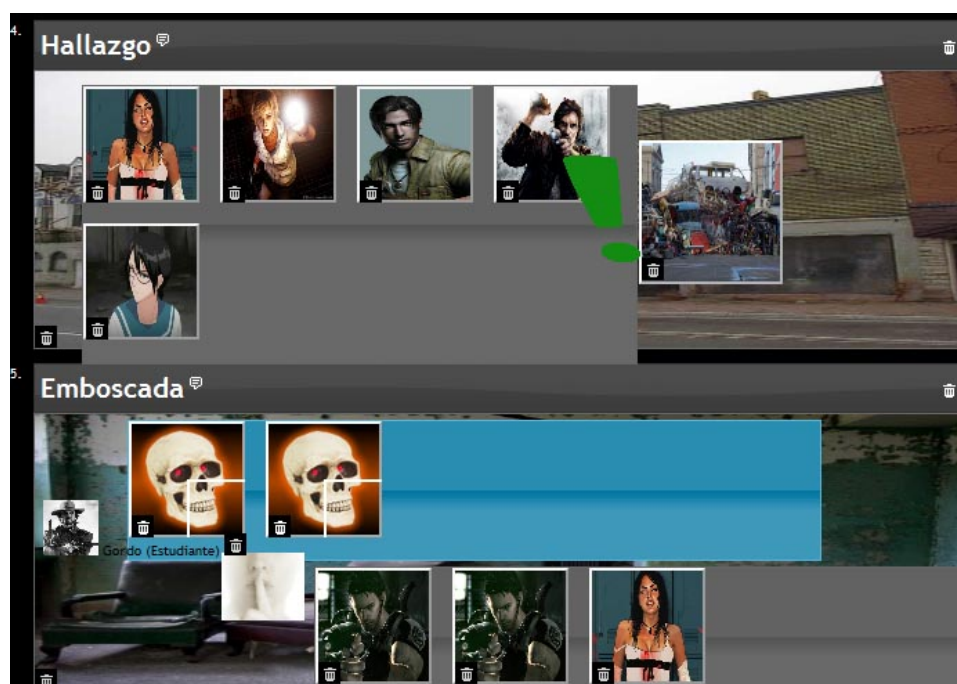


Figura 4.1: Personalización de eventos en *Zombie Story Maker*

la combinación con los demás eventos, siempre ajustándose a la estructura de las historias de temática zombi.

Todos los Eventos a su vez permiten al usuario personalizarlos de alguna manera, ya sea añadiendo personajes como parte activa o pasiva del Evento, objetos o lugares que participan en el mismo. Es perfectamente posible no añadirle a un evento cualquiera de los elementos que permita. Por ejemplo, en el evento de «combate» se puede poner un lugar, indicando que un combate sucedió allí, pero si no se pone, simplemente se sabrá que hubo un combate en ese punto de la historia. La Figura 4.1 ilustra cómo se van personalizando los eventos en *Zombie Story Maker*: las zonas en gris muestran donde se permite agregar el personaje arrastrado (representado en el icono pequeño).

Si el autor es lo suficientemente habil, el lugar puede quedar implícito y no ser necesario añadirlo a este evento. Así mismo, los participantes en el combate, pueden ser omitidos, y de nuevo, pueden quedar implícitos según el diseño del autor. Sobre la implícitud de los elementos de un evento esto es un dato susceptible al estudio: excusando el retruécano, ¿cómo podría designarse explícitamente que algo es implícito? Que algo quede implícito o no, depende en gran parte del contexto de una historia y recae en el conocimiento deductivo o incluso intuitivo que extraemos de la misma. Así pues: un usuario que sea el editor, anotador o diseñador de una historia en el editor, debe indicar cuando algo queda implícito en algún punto de la misma? También

puede quedar indeterminado, debido a que no sea relevante, dependiendo todo del diseño de la historia.

Los Eventos también tienen ciertas restricciones sobre los elementos que permiten, como por ejemplo solo permitir cierto tipo de personajes, o de objetos. La lista completa y descripción detallada de los Eventos (Figura 4.2) es la siguiente:

- **Movimiento:** Este Evento describe un cambio de lugar realizado por alguien, que puede ser uno o más personajes o dejarse implícito. El cambio de lugar puede incluir cualquier lapso de tiempo u omitir cambios de lugar intermedios. Por ejemplo, puede usarse para decir que unos personajes van desde su casa al hospital, y no será necesario indicar que pasan por la calle si no es relevante en la historia. O también puede indicar que están viajando de un lugar a otro (si llegan al destino dependerá del Evento siguiente o no). Este Evento permite tener como protagonista cualquier número de personajes, ya sean humanos o zombis, y permite indicar un lugar de origen y otro de destino.
- **Combate:** Este Evento describe un conflicto físico entre personajes. Permite enfrentar un grupo contra otro y acepta cualquier tipo de personaje. Además permite dotar de armas a los personajes que se quieran, si se estima necesario. El resultado del combate se indicará en Eventos posteriores. Este Evento permite indicar: 2 grupos de cualquier número de personajes de cualquier tipo que se enfrentan uno contra otro, pudiendo usar cualquiera de ellos un arma. También permite indicar el lugar en donde sucede el combate si se desea.
- **Zombis Eliminados:** Este Evento indica que un número de zombis han sido destruidos. Solo acepta cualquier número de personajes de tipo zombi y nada más. El lugar donde sucede queda implícito (por Eventos anteriores donde apareciesen los zombis) o incluso indeterminado.
- **Motín:** Este Evento describe una rebelión de uno o varios personajes contra otro. Los personajes en este Evento deben ser humanos, y pueden llevar armas si se desea: esto describiría una rebelión armada o por la fuerza, y por otra, si no hay armas implicadas puede representar una discusión que conducirá a una separación del grupo (en Eventos posteriores). El Evento admite un grupo (uno o más) de humanos que serán el grupo que se rebela, y un humano, que será el personaje contra el que se rebelan. También se puede indicar el lugar donde sucede.
- **Traición:** Este Evento sirve para indicar que hay una traición entre un grupo de personajes, llevada a cabo por un personaje hacia los demás. Si se desea se permite que el personaje traidor emplee un arma en el

Evento. Los elementos que permite son: un personaje humano que será el traidor, si se desea, un arma que será usada por el traidor, y un grupo de personajes humanos que serán los traicionados. También es posible añadir un escenario al Evento para describir donde sucede la traición.

- Situación romántica: Con este Evento se pueden representar cualquier tipo de situaciones en las que intervengan los sentimientos de los personajes (generalmente de tipo romántico, a cualquier nivel). Permite indicar 2 personajes humanos, que serán los implicados en el romance (aunque con la creatividad suficiente se puede describir una escena de celos).
- Escape: En este Evento se describe una huida o salida de una serie de personajes de un lugar. Permite agregarle cualquier número de personajes de cualquier tipo (los zombis pueden escaparse en el sentido de que los hayan encerrado antes) y un lugar del que huyan o hacia el que huyan. El Evento es lo suficientemente genérico como para permitir esto último, primando generalmente el hecho de que lo que hacen los personajes es salir de un lugar en el que estaban, siendo a veces irrelevante ese lugar o el destino.
- Infección: Este Evento indica que un personaje o más han sido infectados por la enfermedad zombi. El propósito del mismo es otorgar tintes dramáticos, así que el Evento sólo permite agregarle los personajes que se infectan (que obviamente deben ser humanos).
- Fallecimiento: Este Evento indica que un personaje o más mueren en la historia. Al igual que el Evento anterior, sólo se puede indicar los personajes humanos que mueren.
- Zombificación: En este Evento, se indica que un humano ha sido transformado en zombi. Si este Evento debe seguir al Evento de Fallecimiento o no, queda a la discreción del autor de la historia, pudiendo describir si sucede de pronto, que uno de los personajes se convierte en zombi inmediatamente sin que los demás lo sospechasen. Si le precede el fallecimiento, el personaje llevaría un tiempo (descrito en la historia) muerto, y después se transformaría en zombi. Como en los dos Eventos anteriores, éste sólo permite personajes humanos.
- Emboscada: En este Evento, el autor puede representar un ataque por sorpresa o espionaje realizado por un grupo sobre otro. El grupo atacante puede estar compuesto de uno o más personajes de cualquier tipo, que pueden portar armas si se desea. El grupo atacado será uno o más personajes de cualquier tipo, pero no podrán portar objetos

para este Evento. También se puede indicar el lugar donde sucede la emboscada.

- **Atrincheramiento:** Este Evento describe que un grupo de personajes se encierra en un lugar. El Evento permite indicarle un número cualquiera de personajes de cualquier tipo, un lugar que indica donde se han encerrado los personajes, y un objeto de tipo bloqueo, que indicará cómo se han encerrado (si se desea).
- **Hallazgo:** Con este Evento se indica que un grupo de personajes encuentran a otros o a un grupo de objetos. El Evento permite indicar un grupo de uno o más personajes de cualquier tipo, que serán los que se encuentran o encuentran algo, un grupo de objetos de cualquier tipo, que serán los objetos que se han encontrado en el caso de que el Hallazgo sea de objetos, y un lugar que representa dónde se ha producido el Evento.
- **Lograr acceso:** Este Evento describe que un grupo de personajes han logrado acceder a un lugar. El Evento permite indicarle uno o más personajes de cualquier tipo, un lugar al que acceden los mismos, y un objeto de tipo Desbloqueo que mediante lo que han conseguido el Acceso. Si se desea se puede indicar que los personajes llevaban armas para indicar que han entrado al lugar por la fuerza, mediante amenazas, etc.
- **Llegar a:** Por último, este Evento representa la llegada de un número cualquiera de personajes del tipo que sea a cierto lugar (que puede ser indicado o no), por tanto aceptará uno o más personajes de cualquier tipo y un escenario.

Elementos: Además de los Eventos, disponemos de otras características de la historia: los Elementos. Éstos están contenidos en una lista independiente de los Eventos, y se pueden añadir como propiedades a los mismos. Los Elementos son definibles por el usuario y pertenecen a uno de los siguientes tres conjuntos: Personajes, Objetos y Lugares. Los Personajes son todas las personas que aparecen en la historia, ya sean secundarios o protagonistas. En la Figura 4.3 puede verse la pantalla de creación de personajes y, a la izquierda, el menú en donde se almacenarán los que vayamos creando. Los personajes se pueden definir con ciertas características, como por ejemplo el tipo de Personaje, que puede ser zombi o humano, y dentro de los personajes humanos se puede especificar más, como por ejemplo personajes soldado, estudiantes, etc (aunque esto sólo tiene efectos a nivel estético). Además se puede especificar si el Personaje es genérico o no. Esto es importante porque si un Personaje es genérico, cada vez que se añada a un Evento, lo que se está creando realmente es una instancia del mismo, y por tanto, se pueden



Figura 4.2: La lista de eventos tal y como se muestran en Zombie Story Maker

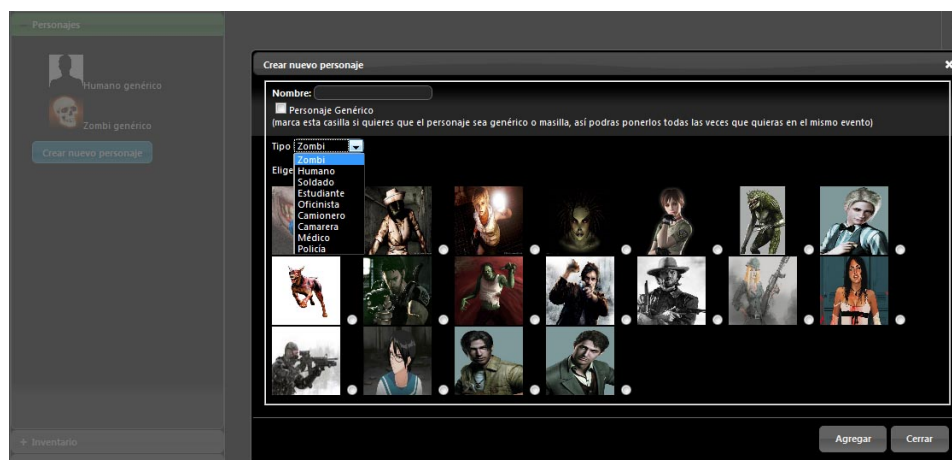


Figura 4.3: Pantalla de creación de personajes

añadir todos las instancias que se quieran de ese Personaje al mismo Evento. En cambio si un Personaje no es genérico, no se instanciará, sino que es el propio Personaje el que se agrega al Evento.

Los Lugares son los escenarios en donde sucederá cada Evento. Como ya establecimos antes, en las películas de zombis el escenario suele pasar a ser un elemento secundario, siendo poco más que un decorado que contiene objetos. En la figura 4.4 puede verse la pantalla de creación de Lugares, y a la izquierda, el menú en donde se almacenarán los que vayamos creando. Algunos eventos no permiten agregarles un escenario, significando esto que no es relevante, y que incluso el escenario puede ser implícito. En esos casos suelen ser Eventos relativos a Personajes en general. Adicionalmente, en los Eventos que permiten asignarles uno o mas Lugares, tambien se pueden omitir, significando lo mismo que en los Eventos sin escenario descritos antes.

Los Objetos, al igual que los Personajes, tambien pueden tener distintos tipos: pueden ser Armas, Bloqueos o Desbloques. Las Armas pueden ser usadas por los personajes en ciertos eventos, generalmente cuando hay que usar la fuerza. En la figura 4.5 puede verse La pantalla de creación de Objetos, y a la izquierda, el menú en donde se almacenarán los que vayamos creando. Los Bloqueos son objetos que se usan para impedir el acceso a un escenario, como pueden ser puertas, barricadas, etc. Por tanto sólo podran participar o aparecer en ciertos tipos de eventos. Los Desbloques son Objetos con la funcion contraria a la de los Bloqueos: permiten el acceso a un lugar. Por tanto, serán objetos como llaves, vehículos, palancas, etc. Al igual que con los Objetos anteriores, esta clase sólo podrá aparecer en los eventos que lo permitan.

Cuando creamos un Objeto, se podrá indicar que ese Objeto es único. Esta propiedad de los Objetos funciona al revés que la propiedad de ser «genérico» de los Personajes. Hemos diseñado esto así porque lo normal

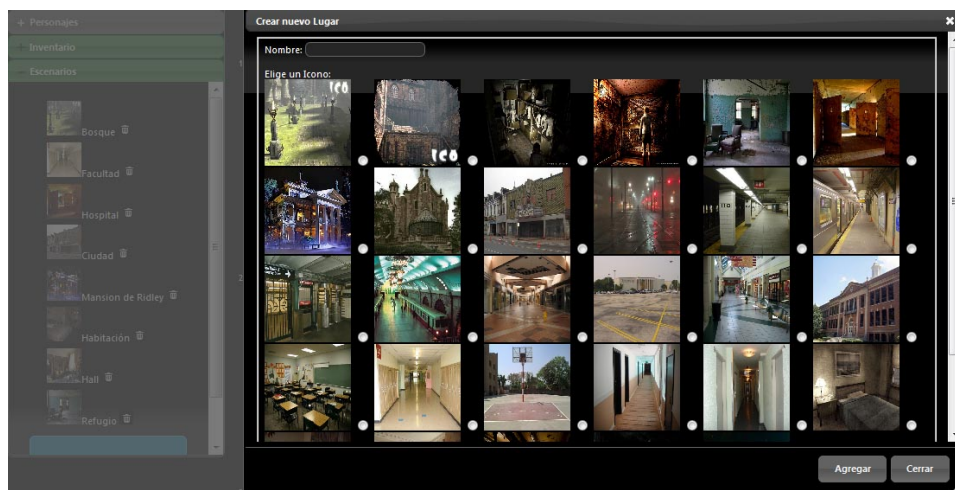


Figura 4.4: Pantalla de creación de lugares

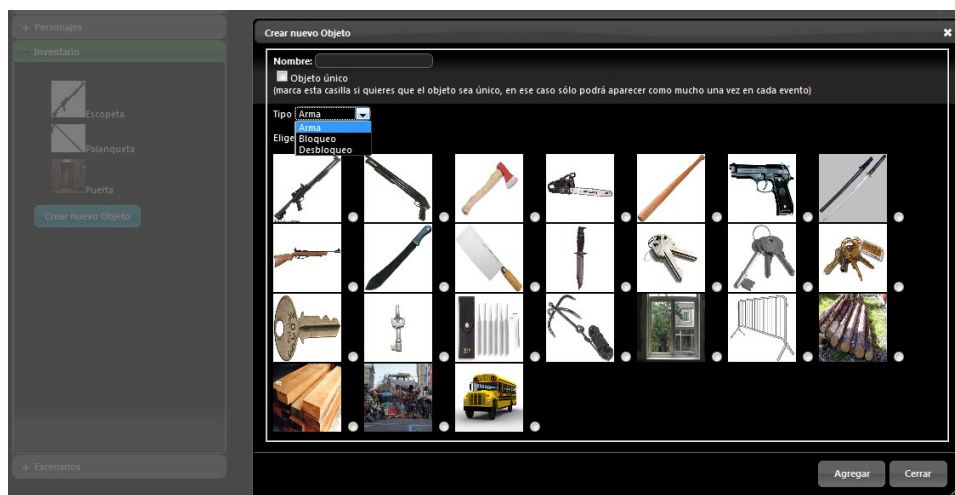


Figura 4.5: La pantalla de creación de objetos

en estas historias es que los Objetos sean genéricos, y por tanto, en nuestro editor lo que aparecerán son instancias de ese Objeto. Si se declara un Objeto como único, será el mismo Objeto el que aparezca en los Eventos que lo contengan, teniendo la limitación que sólo podrá aparecer como máximo una vez en cada uno.

A continuación analizaremos la estructura general y los patrones usados de las historias de temática zombi para que el lector se haga una idea de como modelarlas con la lista de Eventos y Elementos dada.

Las historias de zombis, muy frecuentemente siguen una estructura clásica de narración en tres partes: introducción, nudo y desenlace.

En la introducción, suele presentarse cómo se ha gestado la amenaza zombi y cómo se propaga, y no es excesivamente importante a efectos de lo que ocurre en la historia; la única excepción es que indique el causante de la amenaza zombi. Sin embargo, no es un requisito imprescindible para contar la historia, y en muchos ejemplos reales, como en «El amanecer de los muertos», «Soy leyenda» o «28 días después», es omitida completamente, excepto para presentar a los personajes muy brevemente, llegando enseguida al nudo de la historia. Es interesante sin embargo hacer notar que muchas veces en la introducción se muestran los indicios de la infestación, ayudando a hacerse una idea de que algo no va bien. Como curiosidad, la introducción en este subgénero suele ser muy parecida a la de las historias de desastres (que se puede considerar a su vez un subgénero de otro tipo); en general, las historias de zombis podrían considerarse una historia de desastres de terror, aunque tienen otras características extra que no tienen otras películas con esa mezcla de géneros (como por ejemplo «Aracnofobia» o «Temblores»).

Aunque no todos son obligatorios, los pasos típicos de una introducción son los siguientes:

- Situación en calma delicada: se nos presenta la historia y a los personajes en una situación tranquila. No tardan mucho en verse lo que será el causante de los zombis, que puede ser un laboratorio, un virus o incluso aparecer directamente como zombis ya en marcha, sin mostrarnos la causa clara pero haciéndonos saber que el contagio está en marcha y es cuestión de tiempo que comience la acción. Se muestra por tanto una calma delicada que será rota en cualquier momento.
- Nacimiento del Fenómeno Zombi: Al romperse el equilibrio, el causante del fenómeno zombi es liberado, ya sea porque los zombis que hubiese previamente son avistados (pero no tienen por qué ser reconocidos como tales), el contenedor que mantenía el virus o el gas se rompe, etc. Tras esto, usualmente se dan indicios previos de que algo no va bien, como disturbios, ataques o accidentes esporádicos. Sin embargo los cuerpos de la ley y gobierno hacen poco o nada por contenerlo. Este paso es obligatorio en la historia.

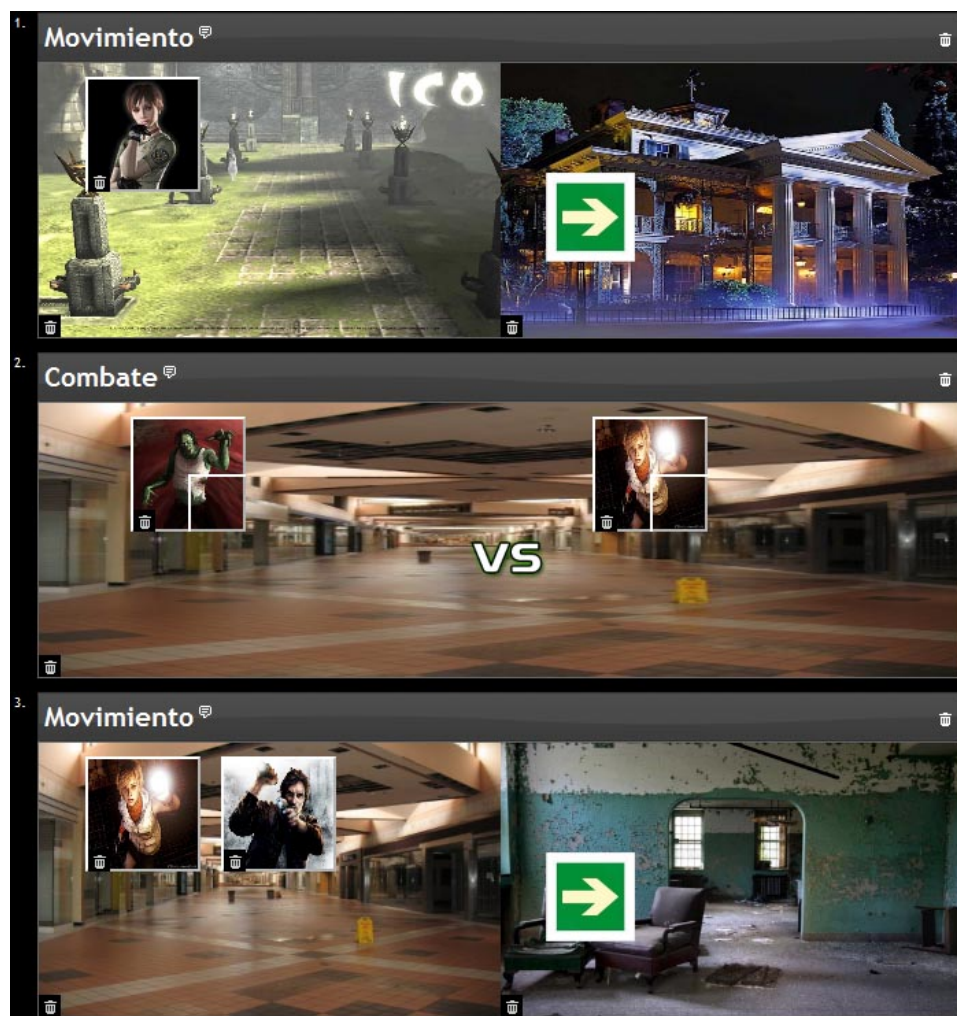


Figura 4.6: Un ejemplo de una introducción: se presenta a los personajes y la primera amenaza.

- **Expansión del Fenómeno Zombi:** Se crean los primeros zombi a partir de humanos corrientes, mostrando que el fenómeno es contagioso (ya sea de una u otra forma, como explicamos antes). Es en este momento cuando los zombis ya son reconocidos como muertos en toda regla dada las evidencias más que suficientes. Los cuerpos de defensa oficiales toman el fenómeno como una amenaza, subestimándolo inicialmente. Este paso puede no suceder explícitamente en la introducción, mostrando como zombis gente previamente viva, por ejemplo. Puede mostrarse explícitamente en el nudo de la historia.
- **Contagios Masivos:** En este punto el número de zombis es enorme y están por todas partes. Los humanos están perdiendo la batalla (si es que la han llegado a presentar) y comienza la trama en sí de la historia.

Como vemos, la introducción nos pone en situación de cómo han surgido los zombis y por qué han conseguido extenderse, siendo una parte bastante homogénea entre todas las historias que hemos analizado.

En este caso y con nuestro modelo, podemos describir cada fase con los siguientes eventos:

- **Situación en calma delicada:** movimiento, hallazgo, traición, llegar a, escape. Típicamente hay pocos eventos relevantes para la historia en este punto, que pueden omitirse perfectamente. El usuario suele describir esto con texto de relleno.
- **Nacimiento del Fenómeno Zombi:** lograr acceso, hallazgo, combate, fallecimiento, zombis eliminados, infección. En esta parte comienzan a suceder cosas, durando hasta la aparición de los primeros zombis.
- **Expansión del Fenómeno Zombi:** combate, fallecimiento, zombificación, zombis eliminados, escape. Esta parte se usa como hemos dicho para mostrar el carácter infeccioso y de invasión del fenómeno, y durará también unos pocos eventos.
- **Contagios Masivos:** es una iteración a mayor escala del punto anterior, situando la acción finalmente en un lugar infestado de zombis.

La siguiente parte de la historia, el nudo, es la que lleva el peso de la trama. Nos presentará un conjunto de personajes con un objetivo, sea éste claro, como llegar a un lugar (Zombieland) o destruir un laboratorio (Resident Evil) o un objetivo difuso, como puede ser simplemente sobrevivir (La noche de los muertos vivientes) o huir (El amanecer de los muertos). En este punto se pueden suceder intervenciones de zombis con desarrollo de personajes y con avances en la trama. Hemos decidido subdividir los sucesos que ocurren en la historia en tres metagrupos, siendo esta clasificación meramente calificativa (no afecta a la aplicación).

- intervenciones de zombis: son la piedra central de la historia y deberían suceder frecuentemente. Ejemplos de estos sucesos pueden ser los ataques de los zombis, los sustos imprevistos, o las trampas que los protagonistas les tiendan. Los eventos que mejor representan estos sucesos son: combate, emboscada, infección, zombificación, escape, hallazgo, fallecimiento y zombis eliminados.
- desarrollo de personajes: aunque no se prodiguen mucho en este tipo de historias, añaden interés a la trama y permiten que sepamos el razonamiento tras las acciones de los personajes. Ejemplos en historias de zombis de estos sucesos son las traiciones o luchas por el poder entre grupos de supervivientes, los saqueos por parte de los personajes, las relaciones personales, etc. Los eventos más significativos para ello son: llegar a, movimiento, escape, situación romántica, traición, motín, emboscada, hallazgo. En algunas historias de temática zombi se hace mucho hincapie en los personajes y sus relaciones, dando énfasis a la falta de moral, miserias humanas o a los errores o inicios de una sociedad, aunque a efectos prácticos no nos interesan estos aspectos y por tanto los dejamos a discrección de que el editor de la historia los añada como texto libre.
- avances en la trama: Suelen darse por un cambio de escenario o encontrar un elemento clave para que los personajes se acerquen o incluso alejen de su objetivo. También pueden servir para situar en el universo de la historia. Por ejemplo, obtener la llave para un lugar, montar en un vehículo y dirigirse a algún sitio, encontrar una pista para una cura, o la inacción por parte del gobierno (a través de comunicados o programas de noticias) representan bien estos sucesos. Los eventos que mejor participan en ello son: hallazgo, atrincheramiento, escape, lograr acceso.

Al ser un subgénero de las historias de terror, tienen un desenlace muy corto, debido a que habitualmente comienza a partir del punto en el que los personajes descubren la razón de lo que está pasando o que objetivo deben cumplir claramente. En las historias de terror, no es extraño que la parte final sea en la que las cosas suceden más rápido y con mayor carga de tensión, tardando poco en cerrarse, dado que como no tienen que acabar bien, sino que normalmente es todo lo contrario, no es necesario dejar cerradas las tramas o trabajarse un final completo para los personajes. A efectos prácticos, la historia puede acabar con cualquiera de los tres tipos de sucesos descritos en el nudo, resolviendo algo o no. Un ejemplo de cada uno de los tres tipos descritos como desenlace serían «Colin» como intervención zombi, «Diary of the dead» como desarrollo de personajes y «Shaun of the dead» como avance en la trama.

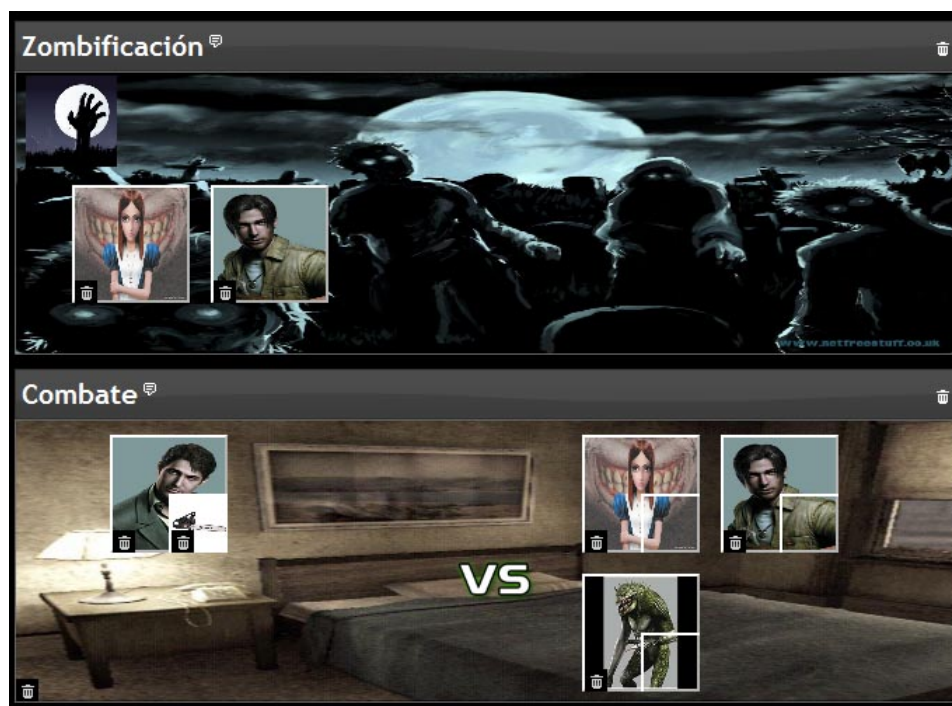


Figura 4.7: Una «escena» de combate contra zombis del nudo de una historia.

4.3. Implementación mediante tecnologías web

En la presente sección vamos a hacer una descripción detallada de la aplicación web que se ha desarrollado, y que demuestra el uso de un editor de historias sencillo y que permite computación cooperativa. La aplicación está colgada en la siguiente dirección y es accesible para todo el mundo, pudiéndose comprobar en todo momento las historias realizadas con ella <http://www.monogatari.es/naum/proyecto/zombisdemo.html>. Está desarrollada usando tecnologías jquery y cuando es necesario, llamadas ajax a páginas php que buscan en base de datos en el servidor. Sin embargo, se tomó la decisión de hacer todo el proceso posible en el cliente, dejando sólo para el servidor tareas como el guardado y carga de las historias. Debemos decir que el objetivo principal de la aplicación era no sólo desarrollar una herramienta para editar historias de zombis, sino que tenía los siguientes requisitos adicionales:

- Debía permitir al usuario anotar las historias.
- Debía ser una herramienta sencilla de utilizar, dando la posibilidad de construir una historia completa en pocos pasos y de forma amena.
- Debe permitir que los usuarios puedan ver y anotar otras historias,

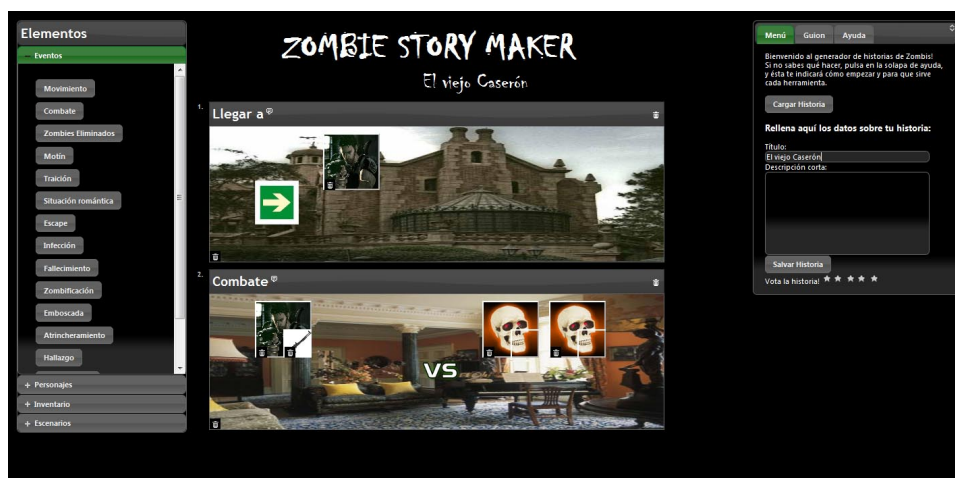


Figura 4.8: Captura de la pantalla principal de la aplicación Zombie Story Maker

mediante computación cooperativa.

El primer requisito es posible mediante la posibilidad de incluir texto libre en cada evento, elegir tipos de personajes complejos, y dotar de una descripción y puntuación a la historia; un usuario podrá así añadir no sólo texto que añada personalización a las historias, sino que además puede calificarlas.

El siguiente punto hace hincapie en la usabilidad y por tanto hemos hecho todo lo posible para que la interfaz de la aplicación sea lo más intuitiva y flexible de utilizar. La siguiente imagen es una captura de la pantalla principal de la web:

Como vemos, la aplicación tiene tres zonas significativas. La primera, en el lado izquierdo, contiene el menú desplegable con todos los elementos que se pueden añadir a la historia. El submenú de eventos permite agregar Eventos a la historia: pulsando en el botón del Evento que deseemos se añadirá al final del guión de la historia.

Cada evento permite la participación distintos tipos de elementos en él: algunos eventos permiten especificar el Lugar en el que suceden, el Personaje que participa, o qué Objetos tienen relevancia. Colocando el ratón sobre el Evento, la aplicación indicará que objeto admite, y poniendo el ratón sobre un objeto ya colocado, mostrarán su nombre. No es obligatorio llenar todos los huecos de un Evento, sólo hace falta rellenar los que tengan sentido para la historia. Por ejemplo, a veces en un combate no hay armas, o un Personaje puede lograr entrar a un Lugar que ya esté abierto o forzando la puerta con un arma. Para personalizar más el Evento, se le puede agregar el texto que se desee pulsando en el pequeño icono de diálogo junto a su título. Puede



Figura 4.9: Un evento recién añadido. Por supuesto, de momento está vacío.

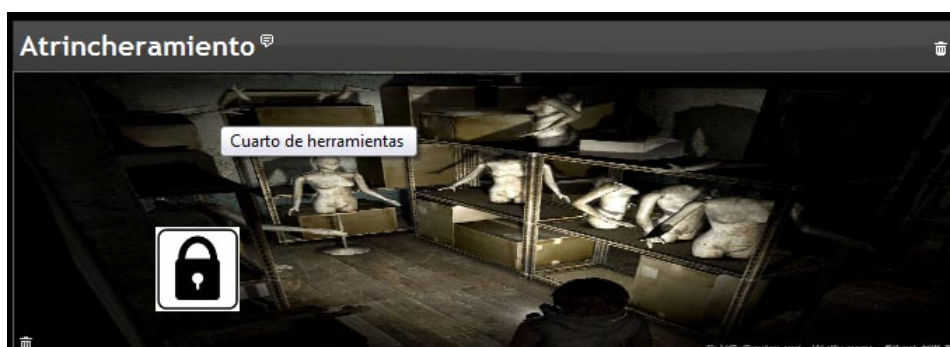


Figura 4.10: Si queremos, podemos añadir al evento anterior un lugar...

ser descriptivo o un diálogo para darle más gracia, descripciones o emoción a la historia. Para borrar un Evento, sólo hay que pulsar en el icono de la papelera que tiene cada uno. Al hacer eso se perderá toda la información que tenía. Los Eventos también pueden ser ordenados entre sí arrastrándolos con el ratón y colocándolos en el orden que se quiera, manteniendo en todo momento los Elementos que tenían.

Los siguientes submenús, el de personajes, el de objetos y el de lugares nos permiten crear los Elementos de la historia que necesitemos y añadirlos arrastrándolos con el ratón a cualquier Evento que lo permita. Además, según la zona que estemos usando de este menú, se actualizará la sección de ayuda de la página.

En el menú de personajes estarán todos los Personajes de tu historia, aunque inicialmente y por defecto hay creados los Personajes del humano y del zombi genérico, para incitar al usuario a usar personajes o a crear los suyos. Para hacer esto último existe un botón que despliega la ventana de crear nuevo Personaje. Cada personaje tiene una clase, y sólo los Personajes que sean humanos (genéricos o de alguna clase humana) pueden realizar ciertos eventos, y lo mismo con los zombis. Se puede crear la versión en zombi

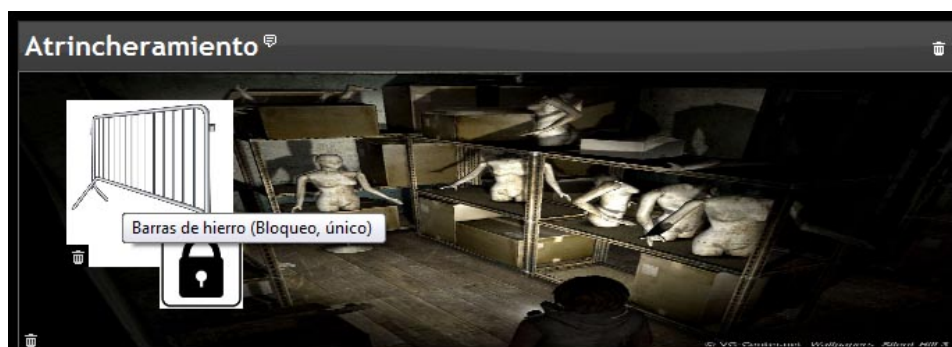


Figura 4.11: Después un objeto. No es obligatorio llenar todos los huecos si no queremos. En este caso queremos especificar cómo han atrincherado el lugar.

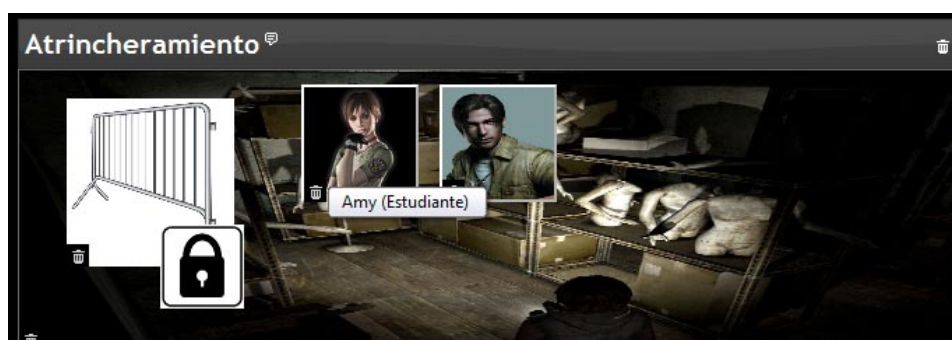


Figura 4.12: Luego decidimos añadir personajes al evento. El orden en el que pongamos cualquier elemento da igual, podíamos haber añadido los personajes al principio si hubiésemos querido.

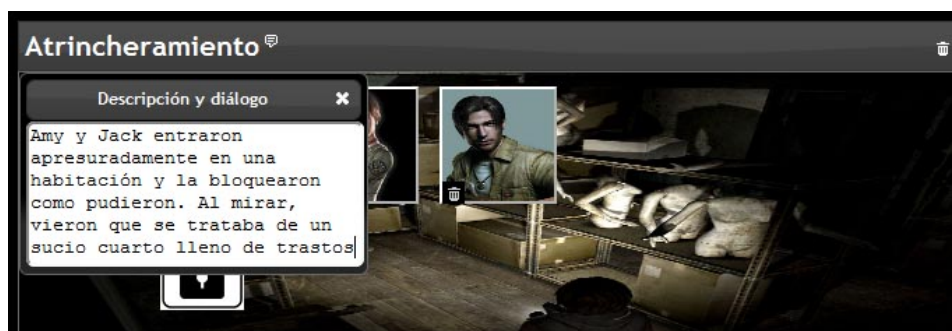


Figura 4.13: Como queremos darle a este evento un toque más personal, añadimos un texto personalizado.

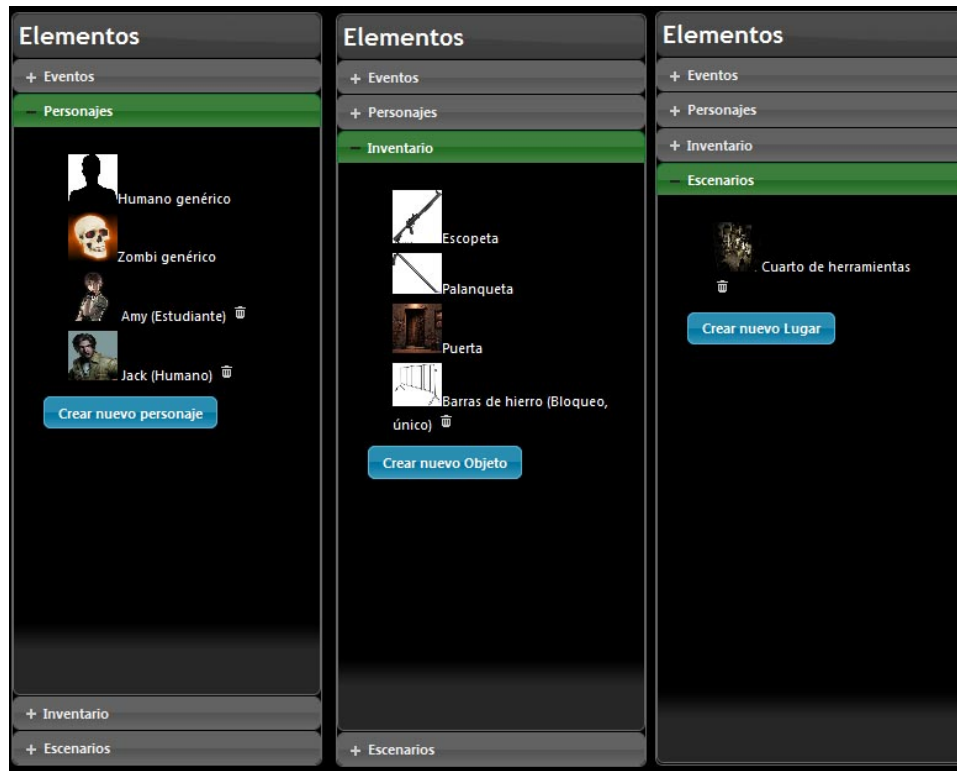


Figura 4.14: Los menús de Personajes, Objetos y Lugares, tal como quedan al desplegarlos en nuestro ejemplo. Pueden observarse los distintos Elementos creados.

de tus personajes humanos si se desea para mostrar distintos estados en el personaje. Para añadir un personaje a un evento sólo es necesario arrastrarlo hasta él: cuando se arrastren, se iluminará la zona en donde los puedes soltar. Una vez añadido un Personaje a un evento, si el Personaje era genérico, se considerará una instanciación del personaje, permitiendo varias copias si se desea para ese evento. En caso contrario, un personaje individual sólo puede aparecer como mucho una vez en cada evento. En ocasiones, al introducir un Personaje en un Evento, se permitirá darle un objeto. Esto significa que el Personaje participa en ese evento con ese objeto concreto. Para dárselo, sólo hay que arrastrar el objeto a la pequeña casilla de la esquina inferior derecha del personaje. Por último, se puede eliminar a los Personajes de la lista pulsando el icono de la papelera junto a ellos. También es posible quitar a los personajes de un evento pulsando en el icono de la papelera. En este caso también se borrarán los objetos que tenía en esa casilla ese personaje.

Más abajo está el submenú de objetos. En este menú tienes todos los Objetos relevantes que hay en la historia. Hay tres creados como ejemplo, de nuevo para enseñar al usuario algunos de muestra e invitarle a que los use o cree otros. Pueden crearse los que se deseen en el menú de crear, eligiendo su tipo de entre :

- **Armas:** son las armas que encontrarán y usarán los Personajes de la historia. Los Personajes pueden combatir con ellas o forzar la entrada de algunos Lugares, por ejemplo. Se corresponden con el modelo de Objetos de tipo Arma que describíamos antes.
- **Llaves:** son Objetos que abren la entrada a lugares cerrados. No tienen por qué ser sólo llaves, pueden ser palancas, ganzúas, tarjetas, etc. Se corresponden con los Objetos de tipo Desbloqueo descritos en el modelo narrativo.
- **Barricadas:** son los Objetos que permiten cerrar a cal y canto un lugar o que se interponen entre un personaje y sus enemigos, por ejemplo. Otro ejemplo es que una llave no es una barricada, sino la puerta que cierra ésta. Se corresponden con los Objetos de tipo Bloqueo de nuestro modelo.

Los Objetos se pueden arrastrar de igual forma que los Personajes, y sus casillas permitidas se iluminarán. Hay restricciones sobre qué tipo de Objeto puedes dejar en ciertos huecos, y si se intenta colocar un Objeto en algún lugar prohibido, el programa avisará de ello. Para eliminar un Objeto de un Evento o Personaje, sólo hay que pulsar en el icono de la papelera que contienen.

El último submenú de esta zona es el de Escenarios. En él, se encuentran todos los Lugares de la historia. Se pueden crear todos los que se quieran

pulsando el botón de crear, al igual que los anteriores Elementos. Tras crearlos, podrán usarse arrastrándolos después a un Evento para indicar que ese Evento sucede en ese Lugar. A veces algún Evento permitirá agregarle dos Lugares, indicando alguna clase de interacción en ellos. También puede que no se permita agregarle ninguno: esto indica que el Lugar donde sucede queda implícito dado el guión de la historia. No es necesario agregar un escenario a todos los Eventos, sino sólo cuando se quiera indicar explícitamente el lugar. Como en los anteriores Elementos, se pueden eliminar lugares de un evento pulsando en el icono de la papelera.

Yendo al lado contrario, tenemos el menú de opciones en la parte derecha de la página. Se puede plegar y desplegar este menú para mejorar la visibilidad de la página si así lo deseamos. Este es el único menú que puede ocultarse del todo puesto que no es imprescindible para crear una historia. En este menú encontramos tres solapas: la de opciones de la historia, que nos permite darle título y descripción a nuestra historia, permitiendo guardarla y puntuarla también, así como ver y cargar cualquiera de las otras historias que haya creadas. Para guardar una historia sólo es necesario darle un título y pulsar el botón de Guardar. En el caso de que hubiese una ocn ese título se sobrescribirá, pero siempre se mantendrá una copia de seguridad en el servidor para poder recuperarla o realizar estudios sobre el proceso de modificación de las historias. Pulsando el botón de Cargar historia se nos mostrará una ventana ocn todas la shistorias disponibles. Sólo es necesario escoger una y la aplicación la cargará automáticamente. Por último, para puntuar una historia sólo hay que pulsar en el número de estrellas que deseemos darle. Una vez cargada cualquier historia se mostrará automáticamente el numero medio de puntuación sobre 5 (redondeada) iluminando el número de estrellas resultante.

La siguiente solapa contiene un resumen de la historia que estamos creando, con el guión de eventos y las notas que les hayamos dado a cada uno. Esto nos permite dar un repaso rápido a la historia completa de un sólo vistazo. Por último, la tercera solapa contiene una ayuda contextual, que nos indica qué podemos hacer con el elemento que estemos usando actualmente.

Finalmente, en el centro se situará la parte más importante de la aplicación: la historia que estamos creando. Es en esta zona donde se añadirán los Eventos cuando pulsemos alguno de los botones de Eventos, y en donde podremos arrastrar los Elementos que hayamos creado para la historia. También aparecerá arriba el título de la historia que le hayamos dado, junto al nombre de la aplicación.

Como nota a pie de página, sobre la accesibilidad de la herramienta, se han añadido medidas estándar como mostrar el título y propiedades de los elementos al situar encima el ratón, y avisos de alerta cuando se intente colocar algún elemento prohibido. Sin embargo, es necesario ver la aplicación en algún navegador que permita ejecución de jquery 1.4 (actualmente prácti-

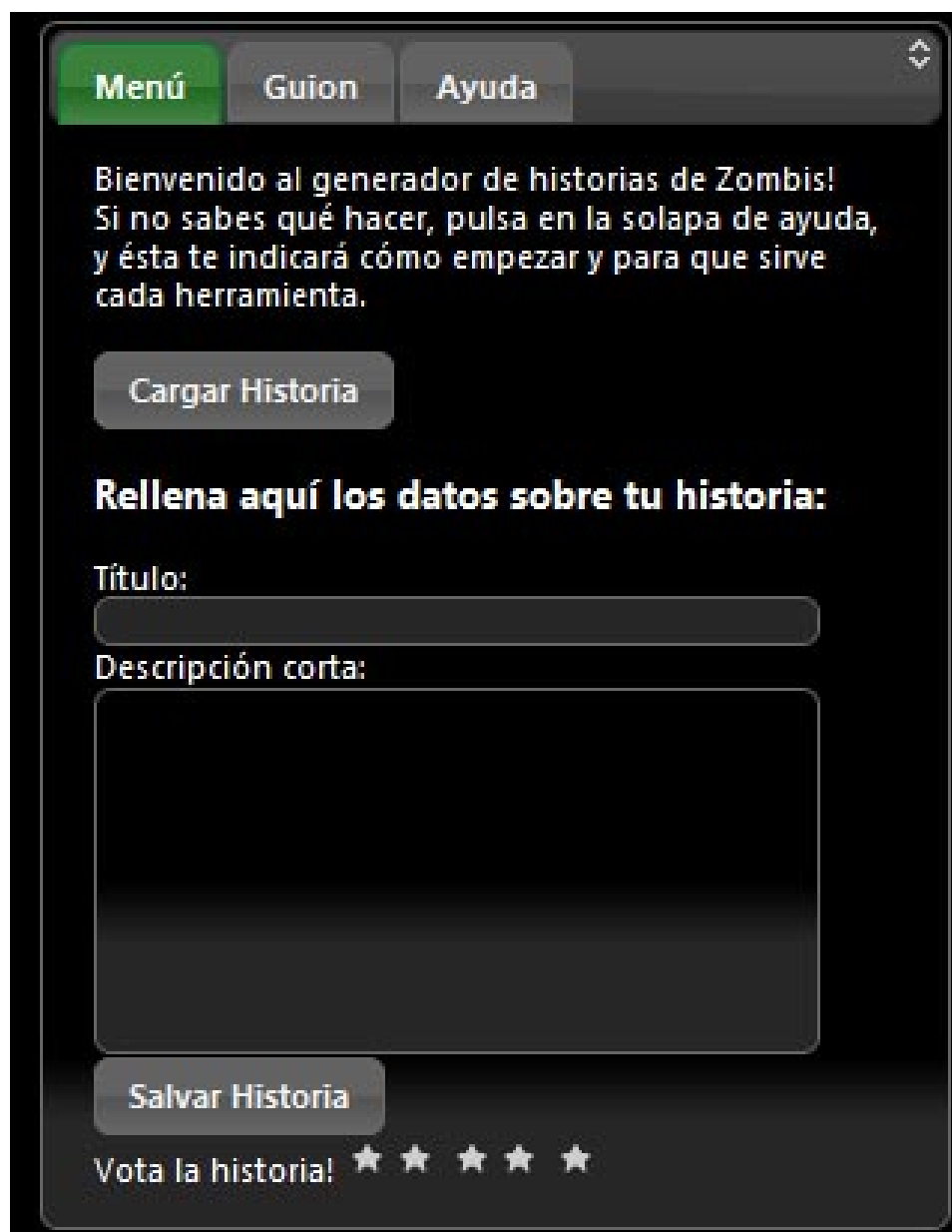


Figura 4.15: Menú principal de nuestra aplicación.

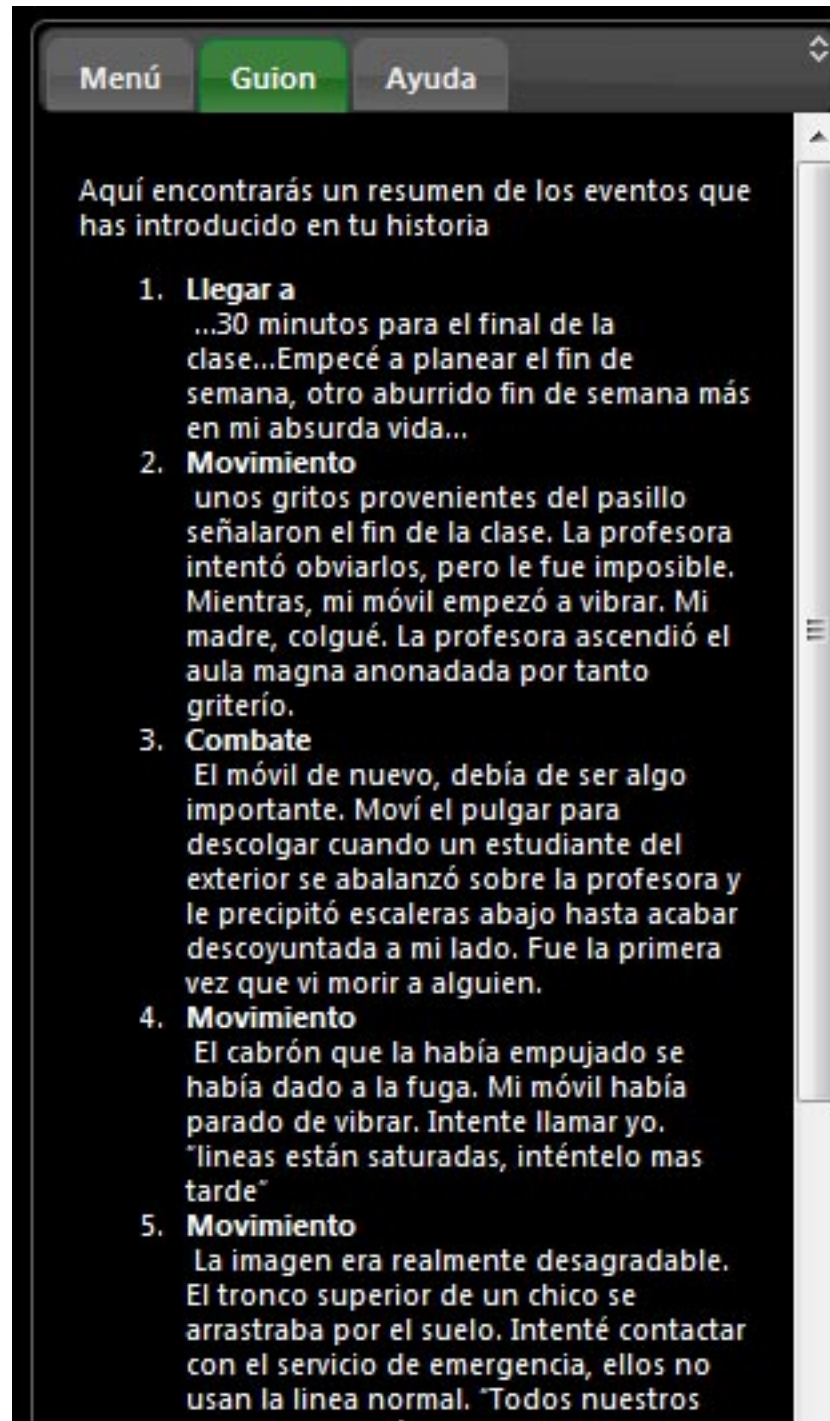


Figura 4.16: Ejemplo de la pantalla de resumen de una historia.

camente todos los navegadores pueden hacerlo, salvo algunos que están ya abandonados, como Internet Explorer 6).

Una vez puesta a disposición del público nuestra aplicación y pasado un tiempo, extrajimos las historias para analizarlas, obtener resultados y compararlas con otras aplicaciones, de lo que se hablará en el siguiente punto.

Capítulo 5

Resultados y discusión

RESUMEN: En este capítulo pasamos a discutir los distintos métodos de evaluación que hicimos sobre nuestra herramienta, y los resultados que obtuvimos, explicando su significado. Después, pasaremos a contrastar nuestro modelo y herramienta con otras aplicaciones de edición narrativa representativas, explicando las ventajas e inconvenientes que pueda tener sobre ellas.

5.1. Análisis del corpus obtenido

Una vez se generó una versión beta de la herramienta y se colgó de internet. Se puso a disposición de la gente en diversos foros y listas de correo. Se les remitió a algunos potenciales usuarios el objetivo de la aplicación y un pequeño cuestionario para valorarla. El correo que se les mandó se presenta en el Apéndice A.

Pero antes poder analizar los resultados y respuestas de nuestra encuesta, debemos describir las métricas que hemos seguido a la hora de evaluar la herramienta y cómo hemos obtenido los resultados. En primer lugar, en el gráfico anterior aparecen los términos historias completas e historias incompletas, pero, ¿qué criterios hemos seguido a la hora de decidir si una historia está completa o no? Lamentablemente no hay una métrica formalizada para ello, o al menos no la hemos encontrado. Sí que hemos considerado aspectos como el concepto de «moraleja de la historia» o «history point», o los patrones que sigue el último acto de las historias de zombis para decidir qué historias están completas y cuáles no. Por supuesto, dado que la herramienta permite modificar o mejorar cualquier historia, una historia completa puede ser ampliada, pero nuestro criterio tiene en cuenta si la historia tiene un final que case con los patrones dados o alcance alguna «moraleja final de la historia» que lamentablemente alude al concepto intuitivo de final, que puede

parecer (y lo es!) arbitrario desde el punto de vista computacional. Así, las historias con contenido pero que no reúnan alguna de las anteriores condiciones serán historias incompletas, y las historias que no tengan contenido alguno (aunque puedan tener eventos), serán consideradas vacías. Además de esto, cuando valoremos las encuestas y comentarios de los usuarios, tomaremos los cuatro parámetros que indicábamos en el correo enviado.

Por otro lado, hemos utilizado para valorar la historia el número de eventos y el número de objetos de cada tipo que tienen. Además, contabilizamos la valoración dada por los usuarios y el número de votos recibidos, con lo que podemos ver el grado de calidad aproximada que tienen las historias. Por último también hemos valorado el número de veces que los usuarios guardaron cada historia, para comprobar cuanto se ha trabajado sobre ella. Para comenzar el análisis, a continuación presentamos un gráfico con el número de historias completas, incompletas y vacías, siguiendo el criterio que mencionábamos antes.



Figura 5.1: Gráfico comparativo entre el número de historias completas, incompletas y vacías

Como vemos, en el gráfico la mayoría de las historias tiene contenido, aunque muchas de ellas no están terminadas. Esto no tiene por qué ser algo malo, sino que en el caso de nuestra herramienta, que permite que varios usuarios trabajen y mejoren sobre otras historias significa que esas historias están dejadas por otro usuario, pero son disponibles para ser modificadas; otros usuarios toman después las historias (tanto las completas como las incompletas) y les añaden elementos y eventos, dejándolas cuando ellos de-

seen. No sólo eso, sino que un usuario será más susceptible de modificar una historia si esta no está terminada. Además, dado que en nuestra aplicación disponemos de un sistema de backup que almacena todas las versiones almacenadas de las historias, podemos ver en cualquier momento los cambios sucesivos que han sufrido y compararlas. En ampliaciones futuras, la comparación estará disponible también para los usuarios, para permitir que ellos sean los que juzguen que versión de las historias les gusta más. Otro dato a tener en cuenta era si los usuarios insertaban mucho texto descriptivo y anotaciones en sus historias. Recordemos que este texto le añade personalidad y «sabor» a las historias, pero no es necesario por sí mismo a la hora de editar y construir una historia narrativa, pues recordemos que nuestro sistema utilizaba los eventos como elemento central y estos tienen como uno de sus objetivos resultar lo más autodescriptivo como sea posible. En el siguiente gráfico, se muestra la proporción de las historias completas e incompletas que tienen texto descriptivo además de sus elementos narrativos constituyentes.



Figura 5.2: Gráfico comparativo entre la proporción de contenido textual entre historias completas e incompletas

Se puede observar que el texto sólo está presente en una pequeña parte de las historias. Un dato interesante es que en proporción, las historias completas con texto son casi la mitad. De igual forma, si contabilizamos las historias completas e incompletas juntas, la proporción es la misma con que con las vacías: algo más de un tercio. La conclusión que podemos sacar es que la decisión de introducir un texto descriptivo no es dependiente de la calidad

o completitud de la historia, sino que más bien es una cuestión personal del autor el decidir añadirlo o no. Sobre la utilidad de este texto descriptivo como anotación, debemos establecer ciertas premisas:

Según Gervás (Gervás, 2010), las metas más deseables acumular por una herramienta de anotación de textos es que incluya identificación sintáctica y semántica, reconocimiento de entidades, solución de ambigüedades y correferencias, identificación cronológica y adición de información intuitiva no explicitada. Nuestra herramienta en principio no tiene un sistema específico de anotación, sino que permite al usuario anotar en texto natural libremente según desee. Esto viene dado por la decisión de que obligar al usuario a especificar ciertos valores podría impedirle realizar la labor de etiquetado ya sea por incapacidad técnica o por desgana. Sin embargo, la actual anotación que permite nuestra herramienta puede ser útil en muchos aspectos. En primer lugar, los textos asociados a los Eventos pueden ser usados como anotaciones semánticas o aclaración sobre el concepto intuitivo del evento. Un motor narrativo que razonase o aprendiese sobre las historias generadas por nuestra herramienta podría extraer esta clase de información asociada al evento. Incluso un análisis sintáctico del texto natural podría indicarnos fácilmente quien es el protagonista de cada Evento. Por otro lado, aunque no sea evidente, podemos usar las historias generadas de forma inversa, esto es, tratar al texto libre como la auténtica historia y la estructura de Eventos y Elementos como notas semánticas sobre cada fragmento de texto, sobre el que podría de nuevo aprender o razonar un motor narrativo.

A continuación, vamos a analizar las historias completas y las incompletas, olvidándonos de momento de las vacías. Como dijimos, las historias incompletas son importantes también: no sólo siguen siendo narrativa al fin y al cabo, sino que son susceptibles de ser completadas progresivamente. Hemos contabilizado los elementos que tienen las historias completas y las incompletas para intentar descubrir qué les falta para ser completas y por qué razón han quedado en ese estado.

A la vista de este gráfico podemos ver que en general los elementos menos usados son los objetos. Ya expusimos en el apartado anterior que los objetos no eran demasiado importantes para la historia en el subgénero zombi, salvo por algún elemento vital de la trama (lo que se conoce como «red herring»). Por otro lado, las historias completas tienen 14'8 eventos de media. Este dato es revelador, ya que a la hora de liberar la web, hicimos dos historias de ejemplo que estaban basadas en dos obras auténticas del género (por tanto, dos ejemplos realistas) que tenían 12 y 15 eventos cada una; por tanto, en cuanto a eventos en la historia, las generadas por los usuarios y que son completas parecen ajustarse en tamaño a las «reales». Mirando a los elementos de la historia, las historias completas tienen 6'2, 3 y 5'5 personajes, eventos y escenarios respectivamente. Las historias de ejemplo que representamos mencionadas anteriormente tienen 9, 2 y 8 una y 5, 2 y 7

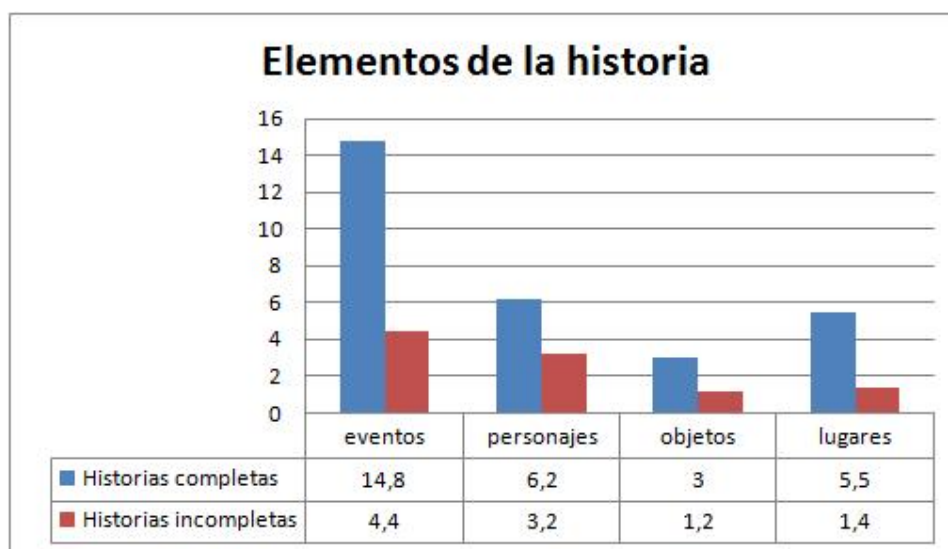


Figura 5.3: Gráfico comparativo entre los elementos narrativos de las historias completas e incompletas

la otra. Lo cual no se aleja demasiado de las muestras aunque es cierto que las creadas por los usuarios tienen menos personajes y lugares en general. La razón parece ser el tipo de historias que cuentan, que en general abarcan escenarios menos amplios, aunque tenemos excepciones que han superado los ejemplos que hicimos. Por otro lado, las historias incompletas vemos que tienen una cantidad menor de objetos y eventos. Por lo general estas historias en su mayoría sólo contienen una introducción narrativa y unos pocos sucesos en su historia, por lo que es lógico que tengan menos elementos que las completas. Para evaluar la calidad de las historias, describimos en la sección anterior un sistema de puntuaciones para valorar las historias. Los valores medios de las historias completas e incompletas se recogen en la siguiente tabla, de la que se puede deducir la calidad de las historias según la opinión de los usuarios:

	Votos	Media Votos/Historia	Media Puntuación
Historias completas	25	2'7	3'51
Historias incompletas	30	3	2'01

Tabla 5.1: Tabla de puntuaciones de las historias

En la tabla, la columna de votos totales indica la suma de los puntos asignados a todas las historias completas e incompletas, la columna de vota-

ciones medias es el número medio de veces que se ha votado cada historia, y la última columna indica el valor medio que tienen ese tipo de historias. Para calcularlo, hemos usado la media del valor de cada historia individualmente. No hemos usado el número total de votos, puesto que hay historias que reciben más y otras menos y por tanto había que calcular su valor por separado. Sin embargo, el número total de votos es indicativo para ver si los usuarios han respondido positivamente al sistema de puntuaciones y si dan muchos o pocos puntos en total dependiendo del tipo de historias. Sin embargo, no debemos olvidar que hay más historias incompletas que completas, y por tanto la media de la primeras es menor aunque en total hayan recibido más puntos: como se puede ver, las historias completas tienen una puntuación media 1'5 puntos mayor que las incompletas (sobre un máximo de 5 puntos). Un dato curioso es que hay algunas historias vacías que están valoradas, y aunque se puede argumentar que las valoró su creador inicial sólo para probar, hay algunas que están valoradas varias veces. Si tenemos en cuenta que la aplicación no permite hacer trampas y valorar de forma repetida una misma historia, es posible que signifique que a los usuarios les parecía prometedor lo que había ya escrito.

A continuación, en las Tablas 5.2 y 5.3, se muestran un extracto de la información obtenida a través de la aplicación. Cada historia se identifica por el título que la ha dado su autor, y de ella conocemos el estado en que se encuentra (*Vacía* porque no tiene eventos, *Incompleta* porque falta por terminar (según el criterio que hemos descrito antes) o *Completa*), si hay texto en lenguaje natural añadido por el autor (Tex), y el número de eventos (Ev), de personajes (P), de objetos (O) y de escenarios (Es) que posee. Además en la segunda tabla mostramos la puntuación media obtenida según los usuarios (Puntos), el número de veces que ha sido votada por estos (Votos) y el número de versiones que ha habido en la historia, según las veces que ha sido guardada (Versiones).

Esta tabla no contiene todas las historias realizadas por los usuarios pero sí una muestra representativa de ellas. A pesar de que el éxito de la herramienta no ha sido masivo, sí que ha contado con una buena base de usuarios y estos han proporcionado un feedback constante. Cuando los usuarios nos enviaron el cuestionario con sus valoraciones, éstas han sido positivas en general, puntuando la aplicación de la siguiente forma según los parámetros que les pedimos:

- Sencillez: 8'75
- Diversión: 7'25

Título	Estado	Texto	Ev	P	O	Es
A Zombie Story	Vacía	N	4	1	0	0
Carla Carla	Incompleta	N	4	3	1	1
Criatura	Completa	S	14	5	2	5
Desayuno con zombies	Incompleta	S	2	2	0	2
Diary of the dead	Completa	N	12	10	3	8
El pequeño traidor	Incompleta	N	4	4	1	2
El zulo maldito	Incompleta	S	8	1	2	0
Fui a por el ordenador y me comieron el cerebro	Incompleta	S	3	2	0	2
La casaaaaaa	Vacía	N	0	0	0	0
La habitación	Vacía	S	0	0	0	0
La matanza	Completa	N	8	4	1	1
Mi primer encuentro	Vacía	N	4	1	0	1
Miedo	Vacía	S	4	0	0	0
Motel Cristina	Incompleta	N	3	3	1	3
Ocaso	Incompleta	S	5	3	0	2
Regustico esquizoide	Completa	S	3	2	1	1
z!	Completa	S	11	5	0	6
ZEPS	Completa	S	11	5	0	4
Zombieland	Completa	N	15	6	3	7
Zombies, sex and chaos	Completa	N	48	14	14	17

Tabla 5.2: Tabla con información estructural de las historias creadas en la aplicación

Título	Puntos	Votos	Versiones
A Zombie Story	2	2	2
Carla Carla	3	3	2
Criatura	5	4	5
Desayuno con zombies	2	3	10
Diary of the dead	3	2	6
El pequeño traidor	0	4	1
El zulo maldito	4	3	4
Fui a por el ordenador y me comieron el cerebro	2	4	6
La casaaaaaa	1	1	1
La habitación	1	1	1
La matanza	2	2	3
Mi primer encuentro	3	3	1
Miedo	1	1	1
Motel Cristina	1	2	2
Ocaso	3	3	1
Regustico esquizoide	5	1	4
z!	4	4	9
ZEPS	4	4	9
Zombieland	2	4	4
Zombies, sex and chaos	4	3	3

Tabla 5.3: Tabla con información sobre la valoración y el uso de las historias creadas en la aplicación

- Interfaz: 8'25
- Rapidez: 7'5

Sin embargo a la hora de dar comentarios, los usuarios fueron más prolijos y nos dieron largos informes para hacer mejoras o quejándose de que no sabían usar la herramienta. A la vista de esto, es posible que los usuarios que se tomaron la molestia de puntuar la herramienta y analizarla ya estuviesen predispuestos a este tipo de tecnologías, editores o páginas web y por ello la puntuaron mejor. Una de las quejas más común de los usuarios es que muchos no sabían que hacer tras añadir los eventos, o en que orden debían editar la historia y añadir elementos (cuando realmente no hay ningún orden establecido). Realmente estos defectos podrían parecer poco importantes dado que no tienen que ver con el modelo narrativo elegido, el dominio o el formato de los datos, pero es importante que tengamos en cuenta que un editor de historias ante todo debe ser usable y todo lo intuitivo que se pueda. En ese aspecto es posible que fuese necesario mejorar la sección de ayuda de la herramienta con algún sistema de tutorial. Otros defectos de interfaz que transmitieron los usuarios fue que era complicado crear historias nuevas después de ver otras, y que la página tardaba en cargar y mostraba todo desmaquetado. Estas mejoras se plantean para realizarse en un futuro a corto plazo. Con respecto al modelo y a la etiquetación, los usuarios principalmente se quejaron de que necesitaban algún evento que indicase un momento de calma. Sin embargo, nuestro modelo narrativo no necesita un evento en el que no suceda nada (mas que diálogo) sino que esto puede suceder en cualquier otro momento. Se considera que si solo hay diálogo y nada más, se puede subsumir ese evento «vacío» al evento anterior o al siguiente. Es posible que el editor en este caso requiriera de mayor habilidad para conseguir estas situaciones, así que también nos hemos planteado introducir alguna clase de eventos nulos que actúen como «azúcar» sintáctico para el usuario, pero que a efectos de modelo, estén contenidos internamente en otro evento más significativo. Otro fallo relevante es que los usuarios necesitaban a menudo saber si se había cambiado su historia, así que sería bueno tener alguna clase de gestión de versiones de la historia, algo que todavía no está disponible, que pudiese indicar los cambios realizados en cada guardado, e incluso bifurcar versiones de una misma historia. Por último, una propuesta que se nos hizo fue la posibilidad de permitir varias líneas temporales de eventos, e incluso permitir una narración no lineal. Es importante decir que actualmente sí es posible incluir varias líneas temporales, tanto líneas paralelas como saltos de tiempo de la historia como flashbacks o fast forwards, pero exige un grado alto de habilidad al usuario que esté creando la historia. Lamentablemente la interfaz no es capaz de mostrar de forma intuitiva estas líneas, y habría que redefinirla, así que eso es lo que propusieron algunos usuarios. En cualquier caso, en historias de zombis no es nada habitual esta estructura de la his-

toria, por lo que inicialmente quedaría considerada como fuera de nuestro dominio.

5.2. Comparación con otros métodos de creación y evaluación

A la hora de comparar nuestra aplicación con otras herramientas, como Scherezade (Elson y McKeown, 2009) o Story workbench (Finlayson, 2008), obtenemos una serie de ventajas e inconvenientes. Como inconvenientes aparece en primer lugar que nuestro dominio es el más acotado de todos. Casi todas las herramientas de edición y anotación tienen dominios generalistas, permitiendo casi todo tipo de historias. Sin embargo, como decíamos en capítulos anteriores esto no tiene pro que ser necesariamente malo, sino que definiendo el dominio de esta forma podemos abarcar todas las posibles historias de forma más completa, sin dejar incoherencias o conflictos en nuestro modelo. Como ventaja, en primer lugar la más importante es que nuestra herramienta es accesible para todo el mundo. Esto hace que cualquiera sea un potencial editor anotador, pero además, al estar en soporte web, hace innecesario mantener instalado cualquier clase de middleware o aplicación para poder usarla, aparte del navegador claro está. Existen aplicaciones de narración que siguen esta filosofía también, aunque no sean demasiadas, y es una decisión que a nuestro parecer es el camino a seguir. Otra de las ventajas que tiene nuestra herramienta es la sencillez de uso. Comparándola con herramientas aceptadas, como Scherezade, Story workbench o la herramienta de Cavazza, nuestro Zombie Story Maker permite construir historias de manera más rápida y sencilla. Obviamente, estamos sólo fijándonos en el proceso de creación y anotación, puesto que nuestra aplicación sólo permite realizar esas tareas. Está claro, que herramientas generalistas, como la de Cavazza o Story Workbench deben ofrecer muchas más opciones y por tanto una complejidad mayor, pero en otras, como Scherezade, aunque tienen un dominio más amplio, adolecen de muchos problemas tanto de interfaz como de complejidad innecesaria. Una de las razones para cerrar el dominio de nuestra aplicación en torno a las historias de zombis, es que eso nos iba a permitir mantener un compromiso entre la sencillez de uso y el alcance que podía tener nuestra aplicación, al limitar las opciones y la personalización a sólo eventos, objetos y situaciones de este género. Para ilustrar este equilibrio, tomemos el ejemplo de scherezade: Permite modelar virtualmente cualquier acción del diccionario (inglés) e integrarla en la historia con las entidades que participen en ella, que también pueden ser virtualmente cualquier concepto de la lengua inglesa. Para organizarlo se alimenta de varias ontologías, permitiendo obtener la momento la clasificación de cada concepto. Sin embargo, esto provoca varios problemas de uso, porque hay que elegir conscientemente todos los eventos correctos y

los elementos con una topología coherente: al leer de WordNet y VerbNet la aplicación da varias opciones al usuario para cada concepto, incluyendo varias que son la misma o muy parecida (esto desconocemos si es por cómo indexa la información WordNet). Esto puede obligar al usuario que tenga que recordar en todo momento que evento «sing» eligió para su historia, o que tipo de «fox» es la instanciación de uno de sus personajes, con lo que aumentan mucho las responsabilidades del autor. Por ejemplo, en la imagen se puede ver la confusión potencial entre términos de WordNet en los editores que usen este corpus.

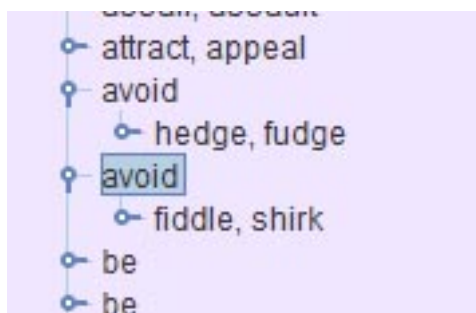


Figura 5.4: Captura de pantalla de la aplicación Sherezade mostrando la línea temporal

Esto sin embargo no es un efecto colateral de tener un dominio tan amplio que permite toda clase de historias, sino de utilizar una ontología de terceros (una decisión completamente lógica y práctica, que es la de aprovechar una base de conocimiento existente y actualizada) que lamentablemente tiene sus inconvenientes. En cambio, tener un dominio tan abierto acarrea otra serie de problemas, como pueden ser la capacidad de restricciones del modelo, que al trabajar sobre una base en principio ilimitada, puede tener problemas de coherencia o completitud. Teniendo un dominio cerrado, como el de las historias de zombis, podemos controlar los eventos y las restricciones que conllevan de una manera mucho más efectiva, sin dejar huecos que nos lleven a incoherencias. Además, otra consecuencia de limitar el dominio es que podemos hacer eventos o elementos más concretos para nuestra historia, permitiendo al usuario pensar a mayor nivel (con una granularidad mas alta, por decirlo de alguna forma). Sin embargo, al limitar el dominio tambien estamos limitando la cantidad de eventos e historias que podremos crear, y debemos tener esto siempre en mente.

Esto nos lleva directamente a otra de las ventajas que tiene nuestro editor: el dar eventos de alto nivel y poco específicos, ayuda a los usuarios a modelar su historia tal como la pensaron, sin hacerles que la desmonten en trozos más pequeños. Sin embargo, a la hora de diseñar nuestro modelo, no creamos todos los eventos con el mismo nivel de granularidad. Algunos eventos como el de «Movimiento» se creo con un nivel más bajo con la inten-

ción de representar situaciones de viaje o en las que el cambio de un lugar a otro fuese importante. Sin embargo, en nuestros resultados podemos ver que algunos usuarios (los de perfil más técnico) lo usaron de manera extensiva para describir todos los movimientos de los personajes, aunque esto ocurre en una minoría de los casos. Sin embargo, la mayoría de los usuarios sí que lo usaron para indicar algún movimiento relevante o algo que sucedía a mitad del camino de un movimiento. Otros eventos están pensados para ser muy amplios y permitir diversos usos al usuario, según la historia que tenga en mente. Dos ejemplos reales vistos en los resultados son por ejemplo «Hallazgo» o «Traición»: En el primer caso, por ejemplo, si se sitúan dos personajes implicados y nada más, el evento describía que dos personajes se encuentran. Por contra, si se añadía un objeto al evento, describía el hallazgo de dicho objeto. En el segundo caso, el evento admite dos grupos de personajes, pero si se incluía sólo uno los usuarios mayoritariamente querían decir que ese era un traidor. Sin embargo, si se incluía un solo grupo de personajes, querían expresar que eran traicionados. Una cosa que queríamos comprobar de qué forma usaría la gente estos eventos «amplios», esto es, si los usarían del modo que pensamos al diseñarlos, o crearían nuevas formas de introducirlos en la historia, y si estas nuevas formas serían coherentes. A la vista de los resultados que obtuvimos, estos ejemplos ponen de manifiesto que si bien surgieron formas no previstas de uso de los eventos, la coherencia de las historias se mantenía. Habría que discutir si esto es beneficioso para el modelo o no, puesto que usar un mismo evento, definiéndose el evento como el elemento atómico de la historia, para distintos propósitos podría ser indicativo de un modelo no formal o incluso difuso. Sin embargo, era nuestra intención mantener al suficiente alto nivel el editor como para que permitiese esta flexibilidad, esto es, permitir diversos usos del evento pero manteniendo el significado semántico del mismo en la historia. El concepto que queremos tomar de «significado semántico» ya fue explicado ampliamente en el capítulo 3, aunque recordemos aquí que pretendía describir que los eventos podían ser usados de manera intuitiva, representando siempre una situación, pero permitiendo al usuario interpretarla (y por tanto describirla) de la manera más flexible posible. Sin embargo, permitir tanta flexibilidad podría provocar que a lo mejor las historias no fuesen lo suficientemente claras. A la vista de nuestro resultado no es el caso, pero de todas formas esa es una de las razones por las que añadimos un campo de texto libre para los eventos, que ayuda a permitir la anotación de las historias y además añade personalidad a cada creación. Con respecto a este campo de texto libre, lo usaron un 28'5 %, aunque para poder contrastarlo con propiedad, podemos ver que porcentaje de historias completas y incompletas tienen texto.

Volviendo a las comparativas, otras dificultades que encontramos en varios editores no sólo en los que hemos puesto de ejemplo, Scherezade e Story workbench sino otros como la herramienta de Cavazza, es que es realmente

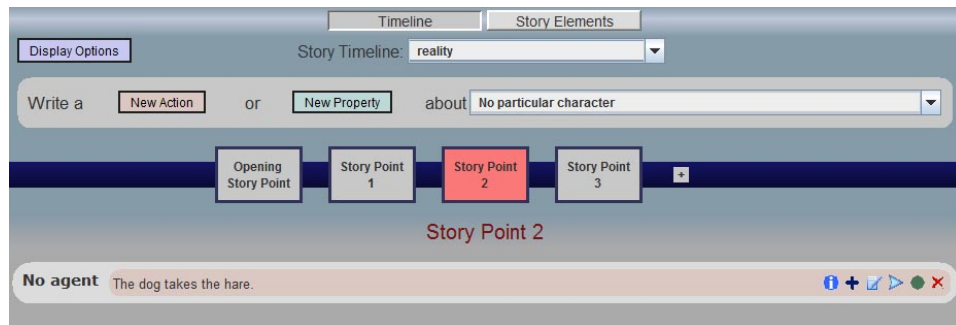


Figura 5.5: Captura de pantalla de la aplicación Sherezade, mostrando eventos en la línea temporal

difícil para un anotador o editor no entrenado el poder realizar la historia y encontrar donde están todas las opciones disponibles para ello. En el caso de Scherezade y otros, hay que navegar por varias pantallas para crear distintos elementos, perdiendo de vista los datos anteriores. Incluso en algunas, no se puede ver la información de todos los pasos de la historia a la vez. En la imagen podemos ver este problema en Scherezade:

Una vez analizados todos estos resultados y a la vista de las estadísticas y comentarios, y tras el análisis comparativo con otras herramientas, podemos pasar a establecer nuestras conclusiones en el siguiente capítulo.

Capítulo 6

Conclusiones

RESUMEN: En este capítulo se analizan los resultados que obtuvimos, sacando conclusiones, se hace un resumen las contribuciones principales de este trabajo y se exponen brevemente las líneas de trabajo futuro que consideramos importante explorar.

6.1. Contribuciones

A la vista de los datos que hemos obtenido de nuestra herramienta de ejemplo, podemos concluir que si bien no mostramos una herramienta revolucionaria para crear historias narrativas, sí que resulta una gran innovación en su aspecto de editor y anotador de corpus narrativos, principalmente debido a dos razones: su facilidad de acceso y uso y la flexibilidad del modelo, y por tanto es más ventajoso en estos campos con respecto a otras que tienen el mismo cometido.

En primer lugar, sobre la facilidad de acceso, hemos demostrado que no es difícil adaptar las técnicas y teorías de la computación de multitudes (crowd computing) y de la web 2.0 a este tipo de herramientas, permitiendo que no sólo cualquier usuario de la web sea un editor o anotador potencial, sino además, nos permite también que las historias del corpus mejoren con el tiempo y sean valoradas. En el mejor caso, también se conseguiría una sinergia entre anotadores, mejorando sucesivamente una historia alcanzando una anotación cada vez más perfecta. Por otro lado, en cuanto a su usabilidad, hemos analizado el modo de uso de otras herramientas de edición narrativa vigentes y hemos observado que generalmente son complicadas de usar y tienen interfaces poco intuitivas. Para evitar esto, hemos intentado adaptar nuestra aplicación a los estándares de usabilidad web y de editores convencionales, logrando así una mayor sencillez de manejo. Esto era un requisito obligatorio, puesto que si la herramienta no era fácil de usar, al estar enfocada a que cualquier usuario conectado a internet pudiese manejarla, no

tendría la base de usuarios que deseábamos puesto que no podrían usarla. En la mayoría de herramientas actuales, es necesario que el editor conozca a fondo el modelo narrativo y reciba alguna clase de entrenamiento en el uso de la aplicación, y esto era algo que tampoco podíamos (ni queríamos!) permitir. Desde un punto de vista tradicional, no conocer a los encargados de construir la base de conocimiento de la que se alimentará tu herramienta es un defecto, pero con el paradigma web de hoy en día, aporta muchos más beneficios que problemas. Durante el primer mes que la aplicación estuvo disponible al público en el periodo de pruebas, los usuarios participaron en alrededor de 30 historias, algo que si bien no es un número alto, es fácil de superar si mejoramos nuestro editor de la forma que indicaron los usuarios y la publicitamos más, además, 349 personas distintas visitaron la web, con lo que en el aspecto de visor de historias, sí que parece que la aplicación cumplió con creces su cometido.

Páginas-URLs (Top 10) - Lista completa - Página de entrada - Salida				
106 páginas diferentes				
	Accesos	Tamaño medio	Página de entrada	Salida
	1773	288 Bytes	7	14
/Monogatari/nahum/proyecto/list.php	538	1.49 KB	75	87
/Monogatari/nahum/proyecto/estrellas.php	442	1 Bytes	45	81
/Monogatari/nahum/proyecto/zombisdemo.html	349	46.20 KB	129	109
	255	3.70 KB	191	182
/Monogatari/nahum/proyecto/inicio.html	241	1.08 KB	113	76
/Monogatari/nahum/proyecto/save.php	112	19.52 KB	18	14
/Monogatari/_myadviser/new_user.php	103	227 Bytes	5	4
/Monogatari/nahum/proyecto/puntuar.php	83	113 Bytes	8	6
/Monogatari/_myadviser/	82	3.19 KB	44	48
Otros	990	44.83 KB	418	432

Figura 6.1: Datos sobre el número de visitas de usuarios distintos a la aplicación.

En el gráfico se muestra el número de visitas a la página principal de nuestra herramienta Zombie Story Maker («zombisdemo.html»), junto con el número de visitas a otras páginas. Como nota aclarativa se han borrado dos páginas que no tenían que ver con nuestra aplicación y además hemos de indicar que el servidor considera como «número de accesos» a las distintas ip de acceso (desde la misma ip, serían «solicitudes»). Nótese también que la página que carga el menú con las historias para visualizarlas es «list.php», por lo que también se muestra el número aproximado de veces que se cargó una historia.

Quizá la razón de que no fuese tan masiva a la hora de obtener historias (en el contexto de cientos de historias) es que exigía que los usuarios trabajasen, y eso sólo suele tener éxito en foros muy concretos de internet. Sin embargo como vemos, a la hora de «leer» las historias creadas por los usuarios, el éxito de nuestra aplicación fue rotundo, dado que por cada visita a la web hubo 1'5 cargas de historia de media.

La otra ventaja que comentábamos era acerca del modelo usado. Si bien es cierto que nuestro modelo no permite usar dominios amplios o generalistas o bebe de una teoría revolucionaria o universal, una vez acotado un

dominio más pequeño y concreto se convierte en un modelo completo y flexible, y muy adaptable al tipo de herramienta que queríamos implementar. De nuevo, al ser un editor de narrativa construido con la filosofía web 2.0, nuestros usuarios no podían ni debían conocer la estructura de nuestro modelo, ni su razonamiento. Por tanto, debíamos mantener subyacente un modelo lo suficientemente flexible e intuitivo para que dichos usuarios construyesen las historias en base a él de forma natural, y sin hacerlo explícitamente. El modelo que hemos presentado gira entorno al concepto de evento semántico, definido como un suceso de la historia con una lista de elementos participantes y que tiene un significado intuitivo. A la vista de las historias creadas y como vimos en los gráficos del capítulo anterior, la aplicación es lo suficientemente potente como para crear complejos trabajos narrativos, y con un relativo poco esfuerzo según los usuarios. Especialmente, los usuarios usaban muy a menudo la opción de añadir texto libre, con lo que la labor de anotación parece cumplida por la herramienta. Sin embargo, hemos comprobado a la vista de los comentarios de los usuarios, que se puede mejorar más el modo de uso del editor, puesto que había veces que algunos usuarios no eran capaces de averiguar como construir una historia. Sin embargo, esto es achacable sólo al interfaz, puesto que practicamente no hemos recibido quejas sobre el modelo. Es más, en ocasiones hubo usuarios que propusieron nuevos eventos, que si no estaban incluidos era porque podían realizarse con otros, pero que a los usuarios les parecía complicado. Esto nos abre las puertas a una cuestión curiosa: deberíamos hacer al modelo redundante a propósito? Cuando diseñamos el modelo, nos acercamos a un límite en el cual había algunos sucesos que podían representarse de alguna forma por varios eventos distintos, pero mantuvimos esos eventos porque además servían para representar tambien otros sucesos radicalmente distintos. Sin embargo, ¿es lógico incluir un evento que es un subconjunto de otros eventos y que no añade realmente ningun tipo de suceso nuevo a las historias posibles?. Aparentemente los usuarios consideraron que así era, por lo que un camino aceptable parece ser el de hacer la herramienta ligeramente redundante para que sea más agradable a los usuarios (user friendly); aun así, es importante volver a constatar que los usuarios supieron como representar lo que tenían en mente con los eventos disponibles y que valoraron muy positivamente la herramienta en cuanto a la rapidez con la que se podían construir las historias que deseaban así que nuestro modelo tambien cumple la labor de editor de historias completo dentro de su dominio.

Además de estas ventajas, y despues de haber analizado los resultados, no sólo nuestra aplicación de creación y edición de corpus narrativos obtiene réditos en el campo de la usabilidad, flexibilidad y bas de usuarios, sino que además, las historias creadas son comparables a las de la literatura y el cine del género elegido. Varios usuarios han comentado que la herramienta viene muy bien si el usuarios tiene en mente una historia que desea plasmar

como si de un storyboard se tratase, o de si quieres contar una historia de manera intuitiva, sin tener que escribirla entera o sin dibujar. Viendo las gráficas de resultados del apartado anterior, había un buen número de historias completas, con estructura idéntica a las de las historias de este tema. No sólo eso, muchas de las historias que calificábamos de incompletas, también pueden ser consideradas historias de pleno derecho semejantes a las de capítulos de historias editadas en el mundo real.

6.2. Trabajo futuro

Ahora bien, a pesar de obtener unos resultados prometedores, podemos ver varios caminos obvios por los que ocntinuar para ir mejorando la herramienta en un futuro cercano. En primer lugar, muchos usuarios dejaban muestras de que la herramienta sería mucho más completa si una vez creada la historia, pudiese generarla en versión texto. Independientemente de que nosotros ya habíamos visto que el proceso de generación de historias a partir de un modelo narrativo iba a ser el camino más evidente a seguir, los usuarios se dieron cuenta de que sería una funcionalidad muy provechosa y consecuente. Además de generar la historia en «papel», la aplicación tiene otro camino potencial para explotar, que está oculto a primera vista. Para poder cargar y guardar las historias, nosotros nos aprovechamos de la tecnología XML con lo que no sería descabellado poder hacer un proceso de conversión para que una vez guardada la historia, se pudiese exportar a otras herramientas como podrían ser Scherezade, jBard o incluso herramientas que no sean de narrativa, como TAP. Sólo sería necesario adaptar el modelo al dominio que deseásemos (siempre manteniendo las condiciones que expusimos en el capítulo anterior) y una vez obtenidas la shistorias, pasarlas a través de una conversión de modelo. Aunque ésto no será trivial ni sencillo, parece posible y nutriría las otras aplicaciones de corpus narrativos. Incluso aunque el dominio de nuestra aplicación fuese menor que el dominio completo de la aplicación en cuyo modelo se transformarían las historias, esta premisa sigue siendo válida, simplemente las historias que añadiríamos a la aplicación de destino pretenecerían a un subconjunto de su dominio.

Por otro lado, también se pretende mejorar a corto plazo la gestión de líneas temporales. Actualmente los sucesos de la historia se cuentan de forma lineal, y aunque es cierto que siendo creativo se pueden insertar distintas líneas temporales tales como flashbacks, líneas paralelas o flash forwards, éstas tendrían un lugar fijo en el orden en el que se cuentan, haciendo a la herramienta un poco rígida en este aspecto. Nuestra propuesta de cambio sería poder mantener distintas líneas (representadas como la que se presenta ahora, pero ocultables) mientras que en la línea principal simplemente podríamos poner un evento de referencia a estas líneas, indicando los eventos que contiene esa línea temporal. Añadir esta funcionalidad no tiene una solu-

cion trivial ni directa, porque habría que decidir y analizar si es posible hacer bucles sobre la misma línea temporal o cómo tratar líneas temporales que se afecten entre sí, pero es un objetivo deseable y que mejoraría las opciones que tiene nuestro modelo de abarcar dominios más complejos.

En cuanto a las anotaciones de los corpus, nuestra herramienta a lo mejor no es demasiado específica en cuanto a las anotaciones realizadas o es poco experta en cuanto a calificar los parámetros narrativos como pueden ser el foco, el tiempo y el ritmo, el tipo de narrador, etc. Sin embargo, el sistema que proponemos permite de manera versátil y genérica usar las anotaciones que contienen las historias, permitiendo usar el texto natural como datos semánticos sobre los eventos y luego realizar búsquedas sobre ellos, o usar los Eventos como datos estructurales y narrativos a la hora de analizar las partes de la historia escrita. Ya vimos que nuestro sistema no trataba de forma específica las anotaciones, pero en el caso de querer adaptar nuestra herramienta para que incluya notas con el objetivo de alimentar un sistema de análisis, generación o edición narrativa, se pueden añadir las opciones para anotar los datos narrativos pertinentes, mientras el modelo nos lo permita: por ejemplo, anotar el tiempo en el que suceden los Eventos sería poco factible si no introducimos la mejora anteriormente escrita, pero indicar el papel de cada Personaje en cada Evento si que es una labor de etiquetado trivial de introducir, e incluso se podría hacerlo de forma más intuitiva en nuestro sistema simplemente mejorando el modo de añadir Elementos a un Evento.

A raíz de las posibilidades que nos ofrece internet, no sólo podemos mejorar la herramienta supliendo carencias que tuviese, sino que se nos abren muchas posibilidades una vez que tenemos nuestro sistema colgado en la red y funcionando. Al tener un modelo formalizado para los elementos de nuestras historias, se podría aumentar las posibilidades del Zombie Story Maker para permitir la narración interactiva al estilo de Say Anything, esto es, permitir la participación por turnos entre uno o varios usuarios y el ordenador, que buscaría en su propio corpus (con CBR, por ejemplo) o en las reglas y patrones extraídos del análisis del dominio las posibles continuaciones a los eventos propuestos por los usuarios. Hemos visto que si la aplicación toma apariencia de juego se volvería mucho más atractiva para los usuarios, así que si usásemos esta propuesta para construir un juego en el que los usuarios definen los elementos o los protagonistas de la historia y deciden que acciones tomar de entre un subconjunto de los eventos disponibles, mientras que el sistema les ofrece el resultado o las consecuencias de dichas acciones, tendríamos un generador de historias que además permite la narración interactiva en el momento de su creación.

En este tipo de aplicaciones podríamos incluir nuevas restricciones relativas tanto a la propuesta de narración colaborativas como a la del «juego» de narración interactiva, como por ejemplo, la de necesidad de un consenso

si se desea matar a un personaje en el caso de una wikinovela, lo que provoca una meta-discusión sobre la trama y requiere de alguna clase de foro de discusión. Para la ficción interactiva se pueden introducir restricciones como condiciones de derrota (que tu personaje muera en la historia) o de victoria (conseguir un objetivo, representado por un estado concreto), o restricciones en cuanto a que eventos están disponibles en cada momento; para esto último sería necesario adaptar cada conjunto de restricciones al dominio estudiando sus patrones narrativos, cosa que ya hemos hecho en nuestra aplicación, pero que no ha desembocado en la inclusión de restricciones narrativas actualmente dada la decisión de permitir mayor libertad a los usuarios para crear la historia.

En conclusión, no sólo es importante crear un repositorio grande de corpus narrativos anotados para poder utilizar de manera práctica herramientas narrativas y además permitir evaluarlas, sino que además aprovechar internet en general y la narración colaborativa o computación de masas en concreto para ello nos da la capacidad de obtener un número grande de historias narrativas realistas (o al menos con la garantía de que son historias creadas en la práctica) en menos tiempo y con mayor grado de diversidad que la que cualquier grupo de pruebas o dedicado a alimentar la base de conocimiento de una herramienta narrativa podría crear.

Apéndice A

Zombie Story Maker: Instrucciones de uso

*El problema con Wikipedia es que
funciona en la práctica, no en la teoría.*

Stephen Colbert

RESUMEN: En este apéndice presentamos el correo completo que se les envió a varias personas para invitarles a que probasen la herramienta. después, mostramos el mensaje que mandamos a 2 foros de mensajes de internet.

A.1. Mensaje enviado a los evaluadores

Soy Nahum Alvarez, estudiante de curso de Máster en Investigación Informática, en la rama de inteligencia artificial. Os escribo para proponeros una colaboración como testers o simplemente pedir vuestra´s valoraciones sobre mi proyecto. Mi investigación es sobre los editores de historias narrativas, y como ejemplo práctico para demostrar mis ideas he desarrollado una aplicación web que consiste en un editor de historias de zombis.

¿Por qué historias de zombis? Necesitaba un editor que fuese prototípico pero que pudiese crear historias completas, y las historias de zombis son un subconjunto que se ajusta perfectamente a ello. Para empezar, sólo tienen 2 tipos esenciales de personajes: humanos y zombis, y los lugares de la historia son poco más que escenarios con poca o ninguna relevancia. Los objetos normalmente sólo tienen 2 ó 3 tipos de usos (armas, herramientas para abrir puertas o bloquearlas) y por último, las historias de zombies son todas muy

parecidas, pues suceden casi siempre los mismos eventos (con alguna excepción que realmente es una película de otro género pero que contiene zombis). Por eso el editor que he hecho permite hacer historias de este tipo.

Ahora, para continuar mi investigación, necesito una base de historias hechas, para lo que necesito colaboración. No sería realista ni podría obtener resultados si las escribiese todas yo, porque además, intento estudiar el efecto que tiene el crowd computing sobre la creación de corpus de historias (lo que hace la aplicación). Lo he colgado en internet y es de acceso público, así que si teneis un poco de tiempo, podeis echarle un vistazo y crear alguna pequeña historia, o mejorar alguna existente, o simplemente leerlas o puntuarlas. Cualquier cosa que hagáis será de utilidad.

Primero pongo la dirección y a continuación explicaré cómo funciona:

<http://www.monogatari.es/nahum/proyecto/inicio.html>

La herramienta es muy sencilla y he intentado hacerla lo más intuitiva posible. Tiene un pequeño menú de ayuda colapsable que explica todo lo que se puede hacer según esté activo un menú u otro. El proceso de creación de una historia es el siguiente: En el menú de la izquierda se encuentran las cuatro opciones con los elementos de la historia. La primera opción contiene los eventos. Básicamente, una historia se compone de una secuencia de estos eventos, a los que posteriormente se les puede añadir elementos. Pulsando en el botón de cada Evento, agregareis uno de ese tipo al final de la historia (aunque luego se pueden reordenar arrastrando con el ratón).

Después, sólo es necesario crear los otros elementos de la historia (personajes, objetos y lugares) que se encuentran en los otros tres menús. Cada menú contiene un botón para crear elementos personalizados y para agregarlos a un evento basta con arrastrarlos con el ratón a donde se desee. Además, cada evento dispone de una ventana de texto libre para añadir comentarios, descripciones, diálogos, etc.

Todos los elementos son eliminables, así que se puede crear y eliminar libremente. Hay que recordar que es un editor, no un generador, así que no hace nada que no se haya construido.

Además de eso, es posible cargar las historias que haga la gente, verlas y puntuarlas con un sistema de estrellas que tiene la aplicación en su menú derecho.

Para colaborar, o proponer ideas u opiniones, podeis escribir a esta dirección de correo: nahum.alvarez.ayerza@gmail.com. Además de los comentarios que deseis hacer, sería de gran ayuda recibir una puntuación de 1 a 10 las siguientes características de la aplicación:

- Sencillez: La herramienta es intuitiva? ¿Permite hacer cosas aparentemente complejas con pocos pasos?
- Diversión: Más que divertir (es un editor, así que no tiene como objetivo divertir, sino crear un storyboard más o menos detallado), ¿la

herramienta ha resultado aburrida(0) o ha sido ameno su uso?

- Interfaz: ¿La apariencia de la aplicación es apropiada? ¿Las opciones y herramientas están visibles y bien colocadas o por el contrario nada era fácil de encontrar y estaba mal maquetado? ¿Parece una herramienta bonita, o parece una web del pasado?
- Rapidez: ¿Crear historias con ella es una tarea pesada o ligera? ¿En pocos pasos ha sido posible conseguir la historia deseada? o por el contrario, ¿fueron necesarios muchos?

Con esos cuatro parámetros podré sacar unas conclusiones detalladas, además de las historias que haga la gente y las puntuaciones de las mismas. Muchas gracias por vuestro tiempo.

A.2. Mensaje enviado al público en general

Estoy haciendo un proyecto de investigación en mi universidad sobre herramientas de creación de historias, y para apoyar mi investigación, he creado un pequeño editor. Digo pequeño, porque sólo se pueden hacer con el un tipo de historias especial, que tienen sólo 2 tipos de personajes esenciales y suceden cosas a menudo comunes en ellas: las historias de zombis.

El editor es una aplicación web «de juguete», suficiente para lo que necesito. Con él se pueden hacer historias sencillas y guardarlas, cargarlas y puntuarlas. Huelga decir que este es un proyecto libre y no es en absoluto publicidad de nada. La ayuda que necesito es que le echeis un vistazo, y si os apetece apetece, cread una pequeña historia (con unos pasos es suficiente, aunque lo que quiera). Bueno, crear, puntuar, comentar, mejorar otras historias o cualquier colaboración sería apreciada. Incluso cambiar el nombre de un personaje o el texto de un evento que os parezca mal descrito. Mi trabajo también trata también en parte sobre el crowd computing, redes colaborativas o networking o como queráis llamarlo. Así que tengo que ponerlo en la red en manos de todo el que quiera usarlo para añadir su granito de arena y comprobar los resultados.

Primero os lo pongo y luego os explico:

www.monogatari.es/nahum/proyecto/inicio.html

La herramienta es muy sencilla y he intentado hacerla lo más intuitiva posible, y de lo que se trata es de crear una secuencia de eventos y añadirle a cada uno elementos.

Solo es necesario crear tus elementos (personajes, objetos y lugares) y arrastrarlos con el ratón a donde queráis. Los eventos son reordenables en la lista y podeis crear y eliminar a vuestro gusto.

Si teneis comentarios o críticas se aceptan, de todo tipo. También si podeis, puntuadme de 1 a 10 los siguientes aspectos:

- Sencillez: La herramienta es intuitiva? ¿Permite hacer cosas aparentemente complejas con pocos pasos?
- Diversión: Más que divertirnos (es un editor, así que no tiene como objetivo divertir, sino crear un storyboard para nosotros mismos), ¿la herramienta os ha resultado aburrida(0) o ha sido ameno su uso?
- Interfaz: ¿Os parece que la apariencia de la aplicación es apropiada? ¿Las opciones y herramientas están visibles y bien colocadas o por el contrario no encontrabais nada y estaba mal maquetado? ¿Os parece una herramienta bonita, o parece una web del pasado?
- Rapidez: ¿Crear historias con ella es una tarea pesada o ligera? ¿En pocos pasos habeis conseguido la historia que queriais? o si no, ¿necesitásteis muchos?

Espero vuestras respuestas.

Apéndice B

Zombie Story Maker: Corpus obtenido

*Para tener éxito, la planificación sola es
insuficiente. Uno debe improvisar
también.*

Isaac Asimov, en Fundación.

RESUMEN: En esta sección listaremos todas las historias completas e incompletas disponibles a día de 01/09/2010, en un formato que incluye su estructura narrativa (eventos y otros elementos), además del texto libre introducido por los autores.

B.1. *Carla Carla*

Personajes:

- Humano genérico
- Zombi genérico
- Carla (Zombi)

Objetos:

- Escopeta

Lugares:

- Sitio

Eventos:

1. Llegar a
2. Combate
3. Zombis Eliminados
4. Fallecimiento

B.2. *Criatura*

Personajes:

- Alfred (Humano)
- Agatha (Humano)
- La Criatura (Zombi)
- Dr Smith (Médico)
- Señor del Camión de la Basura (Camionero)

Objetos:

- Motosierra (Arma)
- Tronco (Desbloqueo)

Lugares:

- Mansión Monfort
- Jardín
- Dormitorio
- Hospital St. Patrick
- Callejón

Eventos:

1. Situación romántica Muchos dirían que Alfred y Agatha eran una pareja feliz. Vivían en una lujosa mansión y esperaban con ilusión su primer hijo. Todo parecía irles bien.
2. Hallazgo. Alfred y Agatha habían decidido recibir a su hijo en casa, y así fue como se hizo. El parto lo atendió el Dr. Smith, quien, al sacar al muchacho del vientre de su madre, enmudeció...

3. Movimiento. Rápidamente hizo trasladar al recién nacido al Hospital St. Patrick, para examinarlo... algo no había salido bien.
4. Infección. El doctor hizo pasar al padre a la sala de operaciones y le mostró el aspecto de su bebé... tenía todo el cuerpo cubierto como de espinas de las que supuraba una sustancia negruzca. Dios mío ¿qué es esto, doctor?
5. Movimiento. Alfred volvió aterrorizado a casa y trató de hacer entender a su mujer que no podía ver al bebé, pero ella le ordenó que lo trajera de vuelta a casa.
6. Traición. Fue entonces cuando Alfred tomó la decisión de decirle a su mujer que el niño había muerto, y deshacerse después de la criatura.
7. Movimiento. Alfred cogió nervioso su coche y se dirigió al hospital... el Dr. Smith no respondía al teléfono.
8. Fallecimiento. Al llegar a la sala donde le había atendido el doctor la encontró abierta y aparentemente saqueada. El cadáver del Dr. Smith yacía en el suelo y no había rastro de la criatura.
9. Emboscada. Al salir al callejón de detrás del Hospital, encontró al señor del camión de la basura tirado entre los cubos. ¡Socorro, algo me ha mordido la pierna, ayuda!
10. Escape. ¡Dios, esto no puede estar pasando! Alfred montó en el coche y huyó de allí en dirección a su casa para alertar a su mujer...
11. Lograr acceso. Al llegar a la casa, el jardín había sido arrasado y la puerta no se abría... a base de golpear con un tronco, logró echarla abajo.
12. Zombificación. Al entrar en el dormitorio de su esposa, Alfred encontró a dos deformes, espinosos y supurantes cuerpos.
13. Hallazgo. En ese momento apareció arrastrándose por la puerta la criatura, que balbuceaba "papáz hacía un ruido ronco, como una risa floja.
14. Combate. Alfred, loco de rabia, agarró la sierra mecánica (que guardaba debajo de la almohada para ocasiones como esta) y comenzó a chillar, maldiciéndolos a todos en nombre de Dios, mientras que los reducía a un amasijo de vísceras. ¡OS ODIO, OS ODIO A TODOS, DEVOLVEDME MI VIDAAARRGG! Y con un movimiento brusco, se segó la cabeza.

B.3. *Desayuno con zombies*

Personajes:

- Howard (Humano)
- Kellog's (Zombi)

Objetos: Ninguno.

Lugares:

- Interior casa familia Stackhouse
- Dormitorio casa familia Stackhouse

Eventos:

1. Movimiento Después de ducharse con su mujer, el señor Stackhouse todavía seguía con hambre... un buen desayuno es lo que necesitaba...
2. Hallazgo Mientras el pequeño Howard degustaba la nueva marca de cereales que le había dejado su madre, cayó en la cuenta de que en la caja no venía ningún regalo... La sorpresa se encontraba detrás de él!!!

B.4. *Diary of the dead*

Personajes:

- Zombi genérico
- Jason (Estudiante)
- Ridley (Estudiante)
- Debra (Estudiante)
- Gordo (Estudiante)
- Tracy (Estudiante)
- Mary (Estudiante)
- Profesor (Humano)
- Grupo superviviente (Humano, genérico)
- Eliot (Humano)

Objetos:

- Escopeta
- Puerta
- Refugio (Bloqueo, único)
- Espada (Arma)

Lugares:

- Bosque
- Facultad
- Hospital
- Ciudad
- Mansion de Ridley
- Habitación
- Hall
- Refugio

Eventos:

1. Movimiento
2. Llegar a
3. Zombificación
4. Hallazgo
5. Emboscada
6. Llegar a
7. Zombificación
8. Fallecimiento
9. Zombificación
10. Combate
11. Fallecimiento
12. Atrincheramiento

B.5. *El pequeño traidor*

Personajes:

- Humano genérico
- Zombi genérico
- La niña fantasma que no muere (Zombi)
- El gafotas traidor (Zombi)

Objetos:

- Escopeta (Arma)

Lugares:

- Camino del cementerio
- Despacho antiguo

Eventos:

1. Llegar a
2. Fallecimiento
3. Escape
4. Combate

B.6. *El zulo maldito*

Personajes:

- Humano genérico
- Zombi genérico
- Gutix (Estudiante)

Objetos:

- Palanqueta
- Puerta
- Bate mágico mata-zombis (Arma)

Lugares: Ninguno.

Eventos:

1. Movimiento Parece que hemos encontrado la entrada a un sitio extraño...
2. Hallazgo ¿Qué es lo que hay en este bote? ¡Oh no! Me ha mordido. Mierda, ¡¡¡está vivo!!!
3. Infección Creo que me ha entrado SIDA, y eso que no comparto las jeringuillas. Me mue...ro... Mecachis.
4. Motín Me encerraré en el armario mugriento, no sea que me salga una cucaracha mutante o algo peligroso.
5. Zombis Eliminados Pum pum pum pum (sonido de vísceras y demás zombieladas).
6. Zombis Eliminados Son muchos zombies Johny!!! ¿Johny???
7. Combate Vaya, mi colega se ha infectado, lucharé con él.
8. Fallecimiento He matado a mi mejor amigo y me queda poco tiempo...
FIN

B.7. *Fui a por el ordenador y me comieron el cerebro*

Personajes:

- Estefy (Humano)
- Antonio Cañete (Humano)

Objetos: Ninguno.

Lugares:

- La casa de Estefy
- El corte ingles

Eventos:

1. Escape Estefy abrazaba a su perro disfuncional mientras lloraba porque se le había roto su portatil. Su madre le dijo que no podía comprarselo hasta septiembre por el tema de la financiacion y pagarlo a plazos, por lo cual ella estuvo muy triste. Entonces decidió buscar ofertas en internet y vio que el Corte Ingles hacía descuentos de hasta el 10 % en

marcas como HP. Decidida llamó a su madre para pedirle la tarjeta e ir a por el ordenador que fuera (llevaba sin conectarse al tienti casi un día!), y su madre le dijo que le dijese a su novio Antony que lo pagase y ella ya le daría el dinero.

2. Situación romántica Asi pues, Estefy decidio ir a casa de su novio para pedirle un adelanto del dinero que necesitaba. Antony, le dio los dineros a su chica y la acerco al Corte Ingles.
3. Lograr acceso

B.8. *La matanza*

Personajes:

- Humano genérico
- Zombi genérico
- Billy (Policía)
- May (Oficinista)

Objetos:

- Escopeta (Arma)

Lugares:

- Station

Eventos:

1. Movimiento
2. Hallazgo
3. Emboscada
4. Infección
5. Escape
6. Situación romántica
7. Combate
8. Zombis Eliminados

B.9. *Motel cristina*

Personajes:

- Kirk (Humano)
- Perro (Zombi)

Objetos:

- Llave (Desbloqueo)

Lugares:

- Habitación
- Laboratorio

Eventos:

1. Emboscada
2. Escape
3. Atrincheramiento

B.10. *Ocaso*

Personajes:

- Marcela (Camarera)
- Serendipity (Policía)
- Lucio, el sucio (Zombi)

Objetos: Ninguno.

Lugares:

- Cancha de baloncesto
- La tasca de Moe

Eventos:

1. Situación romántica - Oh, querida mia, hace cuanto tiempo que no teníamos un momento de intimidad - Marcela se recoge un mechón de pelo que le caía sobre el rostro. - Seren mira de reojo a Marcela, le susurra al oído - Es el cuerpo de policía, ultimamente han pasado cosas realmente extrañas. ¿Sabes cuantos disparos puede aguantar un

cuerpo antes de morir? - Dos o tres - Responde Marcela, mientras deja resbalar la camisa de sus hombros. - ¡Eso pensaba yo también! Pero ayer hubo un cruce de fuego frente al centro comercial con una banda de narcotraficantes y... ¿recuerdas a Pepe, el ayudante de los hombres de Paco? Jura que tuvo que vaciarle 2 cargadores a uno de ellos que olía especialmente mal antes de que cayera abatido. - Es posible que tenga mala puntería.

2. Zombificación Una sombra se cuela en el lavabo de la cafetería y un ronroneo tenue, continuo, marcado se acerca a la puerta del inodoro. Un golpe sordo y un olor putrefacto inunda las fosas nasales de las dos señoritas, una mandíbula rojiza se cierra sobre el cráneo de Serendipity, que pone los ojos en blanco mientras le empiezan a temblar las piernas y sufre violentas convulsiones. Marcela, presa del terror, se revuelve intentando desembarazarse del peso muerto de su amante.
3. Combate Marcela extiende sus brazos palpando a su alrededor buscando frenéticamente algo con que apartar al violento desconocido de Seren. Un martillo, un destornillador, una rail gun... finalmente encuentra junto al retrete los restos de un spray anti mosquitos. Con el mechero de que estaba sosteniendo Seren para quemar la vela sobre ella, la desesperada camarera improvisa un lanzallamas que proyecta una pequeña bola de fuego a la cara del atacante. Tras un pequeño mordisco más en el cráneo de la policía, el ser arde espontáneamente al entrar en contacto con el gas prendido, evaporándose dejando tras de sí una bonita nube rosa.
4. Fallecimiento Totalmente fuera de sí, Marcela incorpora el cuerpo de Serendipity, apoyándolo contra la pared. - ¿SEREN? ¿QUE ERA ESO? ¿ESTAS BIEN SEREN? En ese momento se da cuenta de la mirada perdida de su compañera... y del bulto rosáceo que le sobresale entre los jirones sanguinolentos del pelo. Como impelida por un resorte, Marcela suelta a Serendipity y se incorpora, los ojos muy abiertos, los labios temblorosos.
5. Escape Marcela oye otro ruido en la puerta del baño. Levanta la vista y se encuentra a una docena de clientes del bar quietos, mientras la observan con la misma mirada histérica del desconocido que ha matado a su novia. Un pestañeo y se abalanzan sobre ella. Sin saber muy bien como, con una agilidad felina, Marcela aupa su cuerpo desnudo por la ventana del retrete y sale corriendo a la cancha de baloncesto que hay detrás del restaurante. Según sus pies aterrizan en el pavimento, empieza a correr como alma a la que persigue el diablo...

B.11. *Prueba*

Personajes:

- Zombi genérico
- Bangbang (Estudiante)
- tiabuena1 (Estudiante)
- tiabuena2 (Médico)
- tiabuena3 (Policía)
- Segundón (Camionero)
- superzombie (Zombi)

Objetos:

- Escopeta
- Llave maestra (Desbloqueo)
- Barricada (Bloqueo)
- Machete (Arma)

Lugares:

- Universitat de Lleida
- UdL

Eventos:

1. Atrincheramiento
2. Movimiento
3. Combate
4. Zombis Eliminados
5. Traición
6. Escape

B.12. Prueba 01

Personajes:

- Zombi genérico
- Stewart (Policía)
- Linda (Médico)
- Robert (Soldado)
- Emma (Soldado)

Objetos:

- Escopeta
- Hacha (Arma)
- Machete (Arma)

Lugares:

- Almacén

Eventos:

1. Movimiento
2. Emboscada
3. Combate
4. Zombificación
5. Escape
6. Llegar a
7. Situación romántica
8. Atrincheramiento
9. Infección
10. Traición
11. Combate
12. Zombificación

B.13. *Regustico esquizoide*

Personajes:

- Zardoff (Zombi)
- Belén Esteban (Camionero)

Objetos:

- Metralleta (Arma)

Lugares:

- Torre Pachecus

Eventos:

1. Movimiento Zardoff era un zombie to podrio que algunas noches atacaba a los lugareños de Torre Pachecus con sus chistes cutres.
2. Infección La gente, para hacerle callar le invitaba a tomarse todo el vino Jumillero que tenían...pero era un graso error. Con cada erupción que echaba a la cara las víctimas se convertían en Belenes Esteban.
3. Zombificación Se formaba tal caos en el pueblo que solo una niña pudo frenar la escala de terror friki. -Soy Andreita y no me como el puto pollo mamá- dijo la niña. Todos los habitantes volvieron en sí, e incluso el zombie que se dijo a sí mismo: «Caguendiez, es la última vez que salgo de botellón con Torrebruno y Jesús Gil». Mas tarde se supo que detrás de este asunto estaba la compañía Umbrella-Telecinco, que recababa datos e información para sus estudios raro raro raroos.

B.14. *ZEPS*

Personajes:

- Zombi genérico
- yo (Estudiante)
- el (Estudiante)
- el_2 (Estudiante)
- profesora de sistemas (Camionero)

Objetos: Ninguno.

Lugares:

- clase
- pasillo
- parking
- afueras

Eventos:

1. Llegar a ...30 minutos para el final de la clase...Empecé a planear el fin de semana, otro aburrido fin de semana más en mi absurda vida...
2. Movimiento unos gritos provenientes del pasillo señalaron el fin de la clase. La profesora intentó obviarlos, pero le fue imposible. Mientras, mi móvil empezó a vibrar. Mi madre, colgué. La profesora ascendió el aula magna anonadada por tanto griterío.
3. Combate El móvil de nuevo, debía de ser algo importante. Moví el pulgar para descolgar cuando un estudiante del exterior se abalanzó sobre la profesora y le precipitó escaleras abajo hasta acabar descoyuntada a mi lado. Fue la primera vez que vi morir a alguien.
4. Movimiento El cabrón que la había empujado se había dado a la fuga. Mi móvil había parado de vibrar. Intente llamar yo. «lineas están saturadas, inténtelo mas tarde»
5. Movimiento La imagen era realmente desagradable. El tronco superior de un chico se arrastraba por el suelo. Intenté contactar con el servicio de emergencia, ellos no usan la linea normal. «Todos nuestros operadores están ocupados» Mierda.
6. Llegar a Al llegar al coche me acordé que había quedado con el_2en que le acercaría a casa. «Quédate aquí. Te dejo las llaves, no me lo robes» - Terminé con una costosa sonrisa.
7. Lograr acceso Le vi desde la ventana, y le hice un gesto hacia as escaleras de emergencia. El_2 no paró de hablar, zombies esto, zombies aquello. Le habían alegrado el día.
8. Traición «deje el coche aquí!» Se había dado a la fuga. Si te descuidas te apuñalan por la espalda y... un momento, estaba allí. Conducía el coche hacia nosotros y le perseguían tres fornidos atletas. no parecía que se fueran a cansar. Aceleró
9. Traición Tendría que ser algo rápido, abrir las puertas, subir y... pasó de largo. Es lo que decía de las puñaladas traperas, que nunca...

10. Zombis Eliminados Pegó un frenazo, los anormales que la seguían se estamparon contra mi coche y luego ella dio marcha atrás. Me empezaba a acostumbrar a ver morir gente.
11. Escape Evitar la civilización por el momento era un buen plan...qué demonios! estábamos deseando volver a casa, robar un arsenal y pasarlo de lo lindo!

B.15. *Zombieland*

Personajes:

- Zombi genérico
- Tallahassee (Oficinista)
- Columbus (Estudiante)
- Wichita (Estudiante)
- Little Rock (Estudiante)
- Bill Murray (Humano)

Objetos:

- Escopeta
- Furgon (Arma, único)
- Twinkie (Desbloqueo)

Lugares:

- Carretera
- Columbus
- Supermercado
- Los Angeles
- Mansion de Bill Murray
- Hall
- Parque de atracciones

Eventos:

1. Movimiento
2. Llegar a
3. Traición
4. Hallazgo
5. Combate
6. Movimiento
7. Llegar a
8. Emboscada
9. Fallecimiento
10. Escape
11. Emboscada
12. Combate
13. Zombis Eliminados
14. Situación romántica
15. Hallazgo

B.16. *Zombies, sex and chaos*

Personajes:

- Humano genérico
- Zombi genérico
- Walker (Médico)
- Carrie (Camarera)
- Tampa (Soldado)
- Neo (Oficinista)
- Ghost (Soldado)
- Tiara (Estudiante)
- Connor (Policía)

- Berserk (Zombi)
- Dog (Zombi)
- Nurse (Zombi)
- Zombie (Zombi)
- Kerrigan (Zombi)

Objetos:

- Escopeta
- Palanqueta
- Autobus (Desbloqueo)
- Llaves autobus (Desbloqueo, único)
- Machete (Arma)
- Hacha (Arma)
- Recortada (Arma)
- Llave hospital (Desbloqueo, único)
- Motosierra (Arma)
- Wakizashi (Arma)
- Bloqueo (Bloqueo)
- Puerta cerrada (Bloqueo)
- Glock 17 (Arma)
- C4 (Arma)

Lugares:

- Hospital abandonado
- Jardin
- Calles
- Biblioteca
- Metro
- Entrada metro

- Iglesia
- Alcantarillas
- Hotel
- Pasillo hotel
- Dormitorio Hotel
- Aparcamiento Hotel
- Cementerio
- Aula Biblioteca
- Cuarto Oscuro
- Cuarto Vigilante
- Ciudad infestada de zombies

Eventos:

1. Movimiento
2. Situación romántica zumba zumba zumba zumba ...!
3. Emboscada
4. Escape
5. Combate
6. Fallecimiento
7. Zombis Eliminados
8. Movimiento
9. Hallazgo
10. Movimiento
11. Hallazgo
12. Fallecimiento
13. Escape
14. Movimiento
15. Hallazgo

16. Atrincheramiento
17. Combate
18. Zombis Eliminados
19. Infección
20. Fallecimiento
21. Movimiento
22. Movimiento
23. Situación romántica zumba zumba zumba zumba ...!
24. Zombificación
25. Combate
26. Fallecimiento
27. Movimiento
28. Movimiento
29. Hallazgo
30. Lograr acceso
31. Movimiento
32. Combate
33. Zombis Eliminados
34. Escape
35. Movimiento
36. Hallazgo
37. Movimiento
38. Emboscada
39. Escape
40. Combate BoOoOOooOm!
41. Zombis Eliminados
42. Fallecimiento

43. Situación romántica zumba zumba zumba zumba ...!
44. Emboscada
45. Combate
46. Fallecimiento
47. Zombis Eliminados
48. Llegar a Dead end...

B.17. *historia simple*

Personajes:

- Humano genérico
- Zombi genérico

Objetos: Nignuno.

Lugares:

- Metro

Eventos:

1. Movimiento
2. Combate
3. Zombificación

B.18. *prueba*

Personajes:

- Manolo (Humano)
- Poli diaz (Policía)
- Zombie generica (Zombi)

Objetos:

- Hacha (Arma)

Lugares: Ninguno.

Eventos:

1. Movimiento
2. Fallecimiento
3. Zombificación
4. Combate
5. Infección
6. Situación romántica

B.19. *tret*

Personajes:

- Humano genérico
- Zombi genérico

Objetos:

- Escopeta (Arma)

Lugares: Ninguno.

Eventos:

1. Movimiento
2. Combate
3. Zombis Eliminados

B.20. *z!*

Personajes:

- tetus (Estudiante)
- sesus (Camarera)
- j (Humano)
- vecino misterioso (Zombi)
- sesus (Zombi)

Objetos: Ninguno.

Lugares:

- parque
- mi humilde casa
- portal
- salon
- lavabo (la 2º a la dcha)
- cocina (1º a la izda)

Eventos:

1. Movimiento salgo a correr como todas las mañanas, el barrio mas vacio que de costumbre... «bah! voy a darme prisa, necesito una ducha antes de lleguen éstos...»
2. Combate sesus viene a mi casa como de costumbre (esta vez acompañada) cuando van a entrar al portal alguien se les abalanza encima...
3. Movimiento «tengo un mal presentimiento...» [llaman a la puerta] «mierda» - llega a mi cabeza un estúpido grupo de facebook 'yo también creo que me van a matar cada vez que llaman a la puerta'«¿por qué seré tan cobarde y por qué estaré todo el dia pensando en grupos de fb?»
4. Llegar a «ppfff solo eran éstos» mis colegas llegan a casa. «vaya puto barrio de mierda que tienes! uno de tus vecinos yonkis me ha mordido!»
5. Movimiento «venga, date un poco de agua en la herida esa y deja de quejarte»
6. Movimiento j: «en serio,lo que ha pasado no es normal...» tetus: «no me fastidies j, que crees que pasa...una invasion zombie??ya te gustaría» j: «me gustaría si tuviera en mis manos una recortada...» tetus: «casualmente tengo aquí 3...ah, espera,no...ahi tienes un cuchillo jamonero por si te entra la paranoia...»
7. Infección «oppss esta herida se esta poniendo cada vez mas fea por momentos...»
8. Movimiento «por que sesus jadea tanto?y va golpeando todo? estamos tontos?»
9. Combate «estas bien?! se...eeeh me cago en...» (sesus se tira a mi cuello, intento zafarme pero no lo consigo...que cojones la pasa?y desde cuando es tan fuerte?)

-
10. Zombis Eliminados j se tira encima de nosotras y atraviesa a sesus con el cuchillo/machete que antes le habia mencionado... «eeehh hola?» «estaba infectada! iba a matarte!» «vale j, se te va de manos...acabas de matar a sesus!!» «ya estaba muerta...»
 11. Atrincheramiento tetus: «vale...no me siento cómoda con tu facilidad de asesinar amigos...pero que hacemos? nos encerramos aqui?y si revive?» j: «vale vamos a tranquilizarnos un momento...y a tirar a sesus por la ventana, que me da mal rollo tener un zombie en casa...»

Apéndice C

Zombie Story Maker: Historia de referencia

La lógica es maravillosa, pero a veces obtienes mejores resultados pensando

Terry Pratchett, en El país del fin del mundo.

RESUMEN: Inicialmente, tras elegir la historia que queremos contar y que en este caso está basada en la película *Zombieland*, el proceso para obtener nuestro modelo es sencillo:

1. Primero, vamos creando los eventos que mejor casan con los sucesos de la historia. Los Eventos se mostrarán en el mismo orden que suceden en la historia. En el caso de que nuestra historia no esté basada en nada, podemos irlos creando según se nos vaya ocurriendo el guión.
2. Según vayamos poniendo los eventos, iremos introduciendo en la historia los Elementos que participan en cada suceso, creándolos si no lo habíamos hecho ya, y añadiéndoselos a ese Evento donde creamos oportuno.
3. Cuando acabemos un Evento, se le añade texto libre que consideremos apropiado, indicando diálogos, o descripciones, por ejemplo. Si el evento es lo suficientemente autodescriptivo, puede no ser necesario.
4. Una vez terminados todos los Eventos podemos ponerle título a la historia si no lo hemos hecho ya, y una descripción opcionalmente. Después ya podemos guardarla en la base de datos.

En esta sección incluimos una historia detallada y el proceso de conversión que sufre para adaptarse a nuestro modelo. Para simplificar el apartado, hemos escogido una sin texto libre, pero cuyos eventos son bastante intuitivos.

C.1. Sinopsis de la película *Zombieland*

La película empieza con un estudiante de universidad con pocas dotes sociales y aficionado al World of Warcraft que se dirige hacia Columbus, Ohio para ver si sus padres se encuentra con vida, pero pierde su coche en un accidente y comienza a andar encontrándose por el camino a un cazador de muertos vivientes cuya única determinación en su vida es lograr el último twinkie (un dulce norteamericano). El cazador le dice que no se den sus verdaderos nombres para evitar encariñarse: como el estudiante viaja hacia Columbus, asume ese nombre; como el cazador va hacia Tallahassee, Columbus empieza a llamarle así. Viajan ellos dos solos, pero al parar en un supermercado, se encuentran con dos hermanas, Wichita y Little Rock que les roban el coche junto con el armamento, aunque posteriormente encuentran un Hummer lleno de armas. Más adelante se vuelven a encontrar con las chicas quienes intentan de nuevo robarles el coche, pero Columbus sugiere que viajen juntos, cosa que terminan haciendo.

Columbus le cuenta a Wichita que va hacia Columbus, pero cuando ella comenta que la ciudad ha sido reducida a cenizas, él se da cuenta de que ya no tiene familia a la que acudir. Por ello, decide permanecer con el grupo. Wichita le cuenta que está llevando a Little Rock a «Pacific Playland» en Los Ángeles, un parque de atracciones que se rumorea está libre de zombis.

De camino al parque, pasan a través de Hollywood y Tallahassee decide llevarlos a la mansión de Bill Murray. Tallahassee y Wichita se encuentra con el propio Murray sin infectar, pero caracterizado de zombi para poder pasar desapercibido entre estos. Little Rock no conoce a Bill Murray así que junto con Columbus ambos se ponen a ver la película *Los cazafantasmas*, Murray decide entrar para asustar a Columbus y Little Rock, pero Columbus cree que es un zombi y lo mata.

Mientras tanto Wichita comienza a sentir algo por Columbus, y como no le gusta esa sensación huye con Little Rock a «Pacific Playland». Columbus decide ir tras ellas y Tallahassee se le une en uno de los coches de Murray.

Wichita y Little Rock llegan a «Pacific Playland» y encienden todas las atracciones atrayendo a todos los zombis cercanos. Comienza una batalla en la que las hermanas acaban montadas en una atracción de caída libre y con poca munición. Tallahassee y Columbus llegan justo cuando esta se les está agotando, Tallahassee atrae a la mayoría de los zombis y los asesina, mientras que Columbus salva a las chicas, Wichita le revela su verdadero nombre (Crista) para darle las gracias y se dan su primer beso, tras esto, Columbus se da cuenta que este grupo es la única familia que necesita y los cuatro salen de «Pacific Playland» juntos, no sin que antes Tallahassee disfrute al fin de un twinkie .

C.2. Esquema de la estructura narrativa

Primero, mostraremos los Personajes, Objetos y Lugares de la historia, aunque los fuimos modelando según aparecían en los eventos, sobre todo los objetos y lugares, quedando el siguiente «universo de la historia»: Personajes:

- Zombi genérico (Zombi, genérico)
- Tallahassee (Oficinista)
- Columbus (Estudiante)
- Wichita (Estudiante)
- Little Rock (Estudiante)
- Bill Murray (Humano)

Objetos:

- Escopeta (Arma)
- Furgón (Arma, único)
- Twinkie (Desbloqueo)

Lugares:

- Carretera
- Columbus
- Supermercado
- Los Angeles
- Mansion de Bill Murray
- Hall
- Parque de atracciones

La lista de eventos que utiliza esta historia según su editor, es la siguiente:

1. Movimiento
2. Llegar a
3. Traición
4. Hallazgo

5. Combate
6. Movimiento
7. Llegar a
8. Emboscada
9. Fallecimiento
10. Escape
11. Emboscada
12. Combate
13. Zombis Eliminados
14. Situación romántica
15. Hallazgo

C.3. Formato interno de la estructura narrativa

A continuación mostramos el formato del XML en el que se almacenará la historia:

```
<?xml version="1.0"?>
<ZOMBIESTORYMAKER>
<Elementos>

  <Personajes>
    <item tipo="Humano" generico="S">Humano genérico</item>
    <item tipo="Zombi" generico="S">Zombi genérico</item>
    <item tipo="Oficinista">Tallahasee</item>
    <item tipo="Estudiante">Columbus</item>
    <item tipo="Estudiante">Wichita</item>
    <item tipo="Estudiante">little Rock</item>
    <item tipo="Humano">Bill Murray</item>
  </Personajes>
  <Objetos>
    <item tipo="Arma">Escopeta</item>
    <item tipo="Desbloqueo">Palanqueta</item>
    <item tipo="Bloqueo">Puerta</item>
    <item tipo="Arma" unico="S">Furgon</item>
    <item tipo="Desbloqueo">Twinkie</item>
  </Objetos>
  <Lugares>
    <item>Carretera</item>
```

```

    <item>Columbus</item>
    <item>Supermercado</item>
    <item>Los Angeles</item>
    <item>Mansion de Bill Murray</item>
    <item>Hall</item>
    <item>Parque de atracciones</item>
  </Lugares>
</Elementos>

<titulo>Zombieland</titulo>

<eventos>

  <evento nombre="Movimiento">
    <lugar>Carretera</lugar>
    <lugar>Columbus</lugar>
    <pj tipo="Estudiante" nombre="Columbus"></pj>
    <pj tipo="Oficinista" nombre="Tallahasee"></pj>
  </evento>

  <evento nombre="Llegar a">
    <lugar>Supermercado</lugar>
    <pj tipo="Oficinista" nombre="Tallahasee"></pj>
    <pj tipo="Estudiante" nombre="Columbus"></pj>
    <pj tipo="Estudiante" nombre="Wichita"></pj>
    <pj tipo="Estudiante" nombre="Little Rock"></pj>
  </evento>

  <evento nombre="Traición">
    <lugar>Supermercado</lugar>
    <grupo>
      <pj tipo="Estudiante" nombre="Wichita">
        <obj tipo="Arma">Escopeta</obj>
      </pj>
    </grupo>
    <grupo>
      <pj tipo="Estudiante" nombre="Columbus"></pj>
      <pj tipo="Oficinista" nombre="Tallahasee"></pj>
    </grupo>
  </evento>

  <evento nombre="Hallazgo">
    <lugar>Carretera</lugar>
    <pj tipo="Estudiante" nombre="Columbus"></pj>
    <pj tipo="Oficinista" nombre="Tallahasee"></pj>
    <obj tipo="Arma" unico="S">Furgon</obj>
    <obj tipo="Arma">Escopeta</obj>
  </evento>

```

```

<evento nombre="Combate">
  <lugar>Carretera</lugar>
  <grupo>
    <pj tipo="Estudiante" nombre="Columbus">
      <obj tipo="Arma">Escopeta</obj>
    </pj>
    <pj tipo="Oficinista" nombre="Tallahasee">
      <obj tipo="Arma">Escopeta</obj>
    </pj>
  </grupo>
  <grupo>
    <pj tipo="Estudiante" nombre="Wichita">
      <obj tipo="Arma">Escopeta</obj>
    </pj>
    <pj tipo="Estudiante" nombre="Little Rock">
      <obj tipo="Arma">Escopeta</obj>
    </pj>
  </grupo>
</evento>

<evento nombre="Movimiento">
  <lugar>Carretera</lugar>
  <lugar>Los Angeles</lugar>
  <pj tipo="Oficinista" nombre="Tallahasee"></pj>
  <pj tipo="Estudiante" nombre="Columbus"></pj>
  <pj tipo="Estudiante" nombre="Wichita"></pj>
  <pj tipo="Estudiante" nombre="Little Rock"></pj>
</evento>

<evento nombre="Llegar a">
  <lugar>Mansion de Bill Murray</lugar>
</evento>

<evento nombre="Emboscada">
  <lugar>Hall</lugar>
  <grupo>
    <pj tipo="Humano" nombre="Bill Murray"></pj>
  </grupo>
  <grupo>
    <pj tipo="Oficinista" nombre="Tallahasee"></pj>
    <pj tipo="Estudiante" nombre="Little Rock"></pj>
    <pj tipo="Estudiante" nombre="Wichita"></pj>
    <pj tipo="Estudiante" nombre="Columbus"></pj>
  </grupo>
</evento>

<evento nombre="Fallecimiento">
  <pj tipo="Humano" nombre="Bill Murray"></pj>
</evento>

```



```
<evento nombre=" Situación romántica</label>
  <pj tipo=" Estudiante" nombre=" Columbus"></pj>
  <pj tipo=" Estudiante" nombre=" Wichita"></pj>
</evento>

<evento nombre=" Hallazgo</label>
  <lugar>Parque de atracciones</lugar>
  <pj tipo=" Oficinista" nombre=" Tallahasee"></pj>
  <obj tipo=" Desbloqueo">Twinkie</obj>
</evento>
</eventos>
</ZOMBIESTORYMAKER>
```

C.4. Representación gráfica de la estructura narrativa

En esta sección mostraremos como queda construida la historia en nuestra herramienta. En primer lugar se presentan los menús con las listas de Elementos ya creados. Primero ofreceremos las listas de Personajes, Objetos y Lugares respectivamente, y por último, la de Eventos.



Figura C.1: Lista de Personajes de nuestra historia



Figura C.2: Lista de Objetos de nuestra historia



Figura C.3: Lista de Lugares de nuestra historia

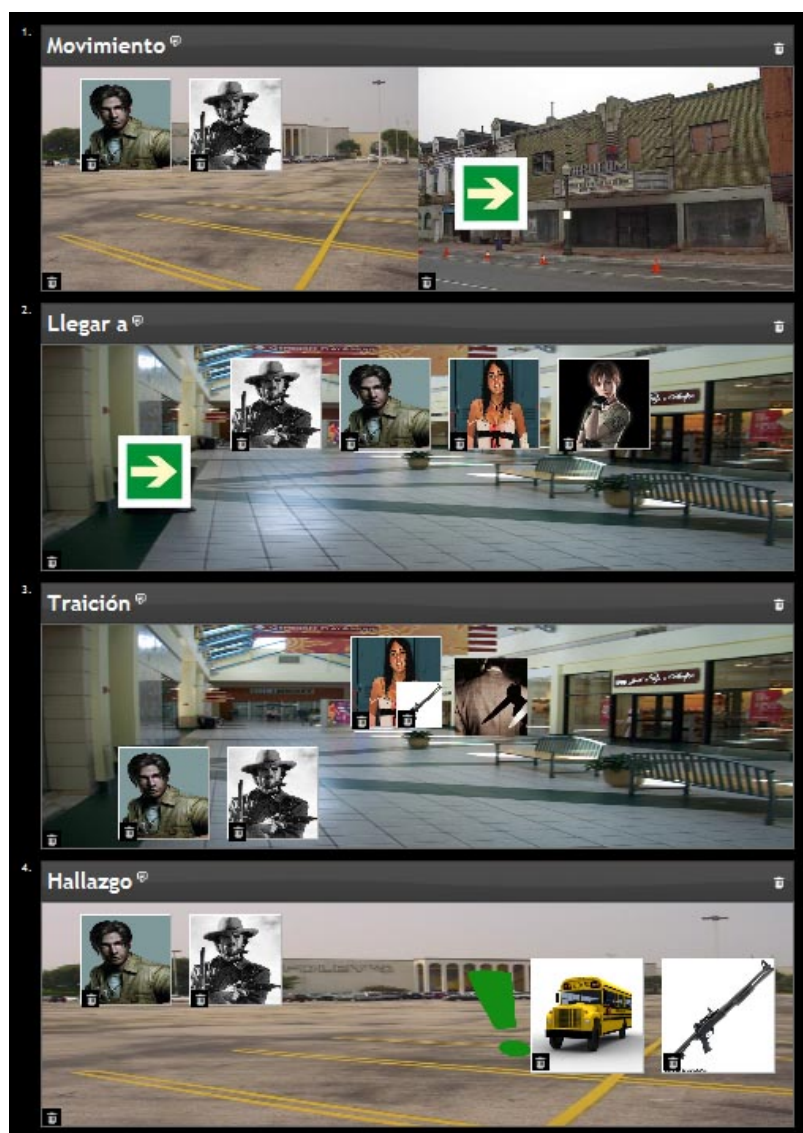


Figura C.4: Lista de Eventos de nuestra historia (1/4)

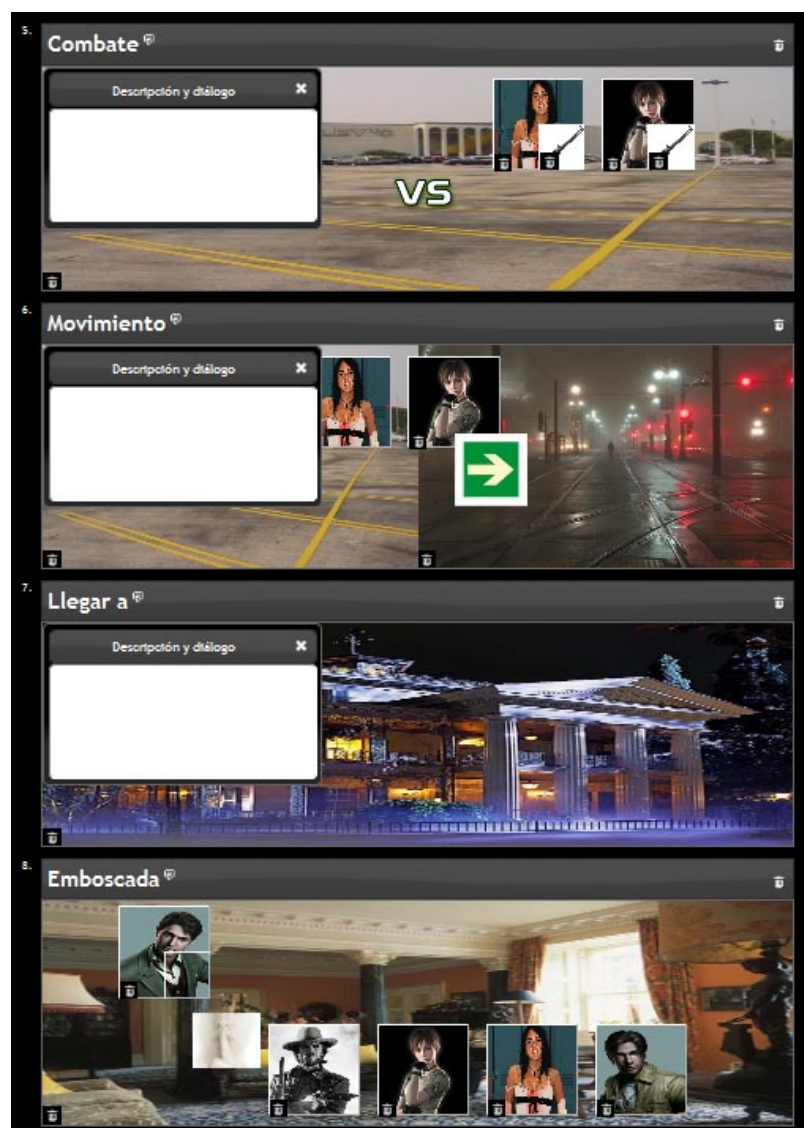


Figura C.5: Lista de Eventos de nuestra historia (2/4)

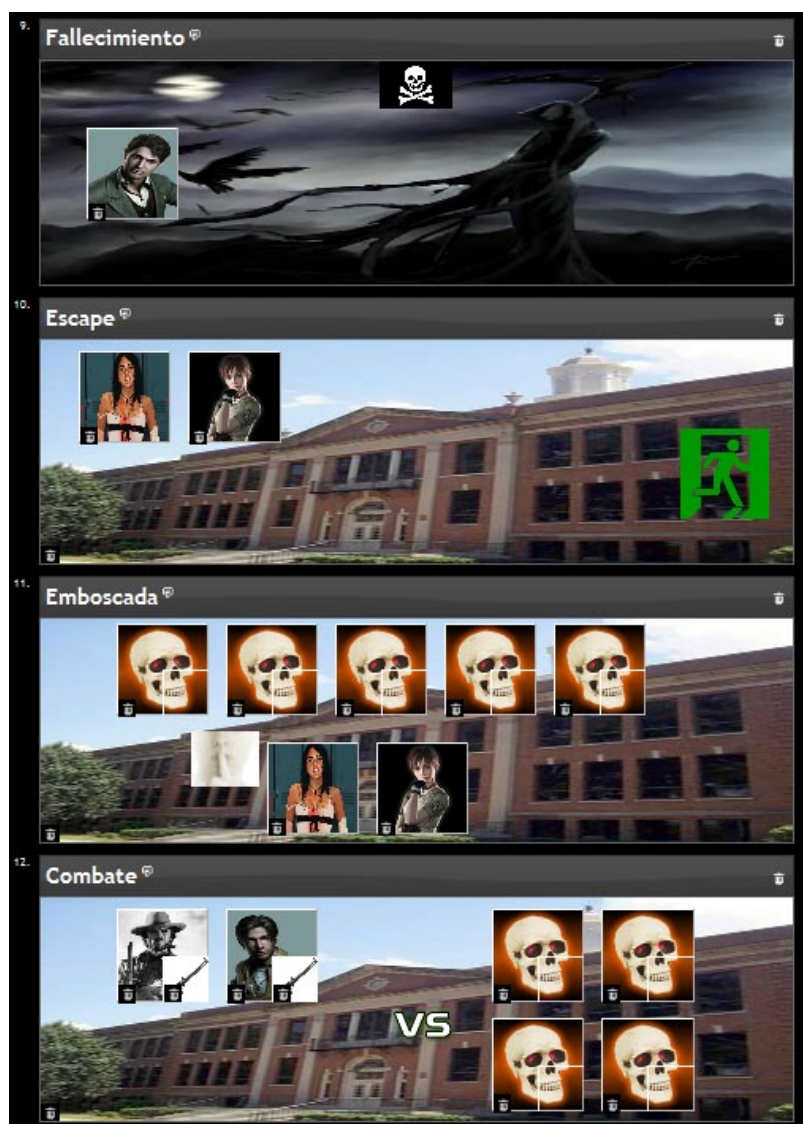


Figura C.6: Lista de Eventos de nuestra historia (3/4)

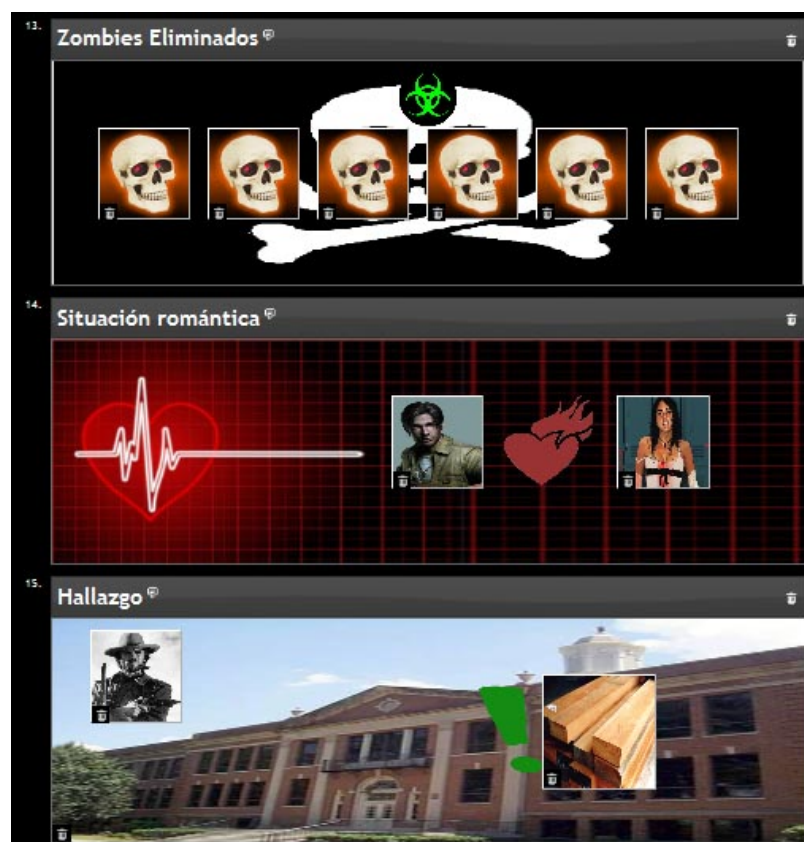


Figura C.7: Lista de Eventos de nuestra historia (4/4)

Bibliografía

*Y así, del poco dormir y del mucho leer,
se le secó el cerebro, de manera que vino
a perder el juicio.*

Miguel de Cervantes.

- BAL, M. *Narratology : introduction to the theory of narrative*. University of Toronto Press, 1997. ISBN 0802078060.
- BRINGSJORD, S. y FERRUCCI, D. A. Artificial intelligence and literary creativity: Inside the mind of brutus, a storytelling machine. Lawrence Erlbaum Associates, Hillsdale, NJ, first edition, 1999.
- CAVAZZA, M., CHAMPAGNAT, R. y LEONARDI, R. The iris network of excellence: Future directions in interactive storytelling. En *ICIDS*, páginas 8–13. 2009.
- CAVAZZA, M., PORTEOUS, J. y CHARLES, F. Narrative generation through character's point of view. En *AAMAS*. 2010.
- CHAMBERS, N. y JURAFSKY, D. A database of narrative schemas. En *LREC*. 2010.
- CORRADINI, A., MEHTA, M. y ONTAÑÓN, S. Evaluation of a drama manager agent for an interactive story-based game. En *ICIDS*, páginas 246–257. 2009.
- DAVIDSON, D. The logical form of action sentences. En *The Logic of Decision and Action*, Rescher, N.. Pittsburgh: University of Pittsburgh Press, 1967.
- ELSON, D. K. y MCKEOWN, K. Building a bank of semantically encoded narratives. En *LREC*. 2010.
- ELSON, D. K. y MCKEOWN, K. R. A tool for deep semantic encoding of narrative texts. En *ACL-IJCNLP '09: Proceedings of the ACL-IJCNLP 2009 Software Demonstrations*, páginas 9–12. Association for Computational Linguistics, Morristown, NJ, USA, 2009.

- FINLAYSON, M. A. Collecting semantics in the wild: The story workbench. En *In*, páginas 46–53. AAAI Press, 2008.
- GALTON, F. *Vox populi*. vol. 75. London, 1907.
- GERVÁS, P. Corpus annotation for narrative generation research. Viena, Austria, 2010.
- GODDARD, C. *Semantic Analysis. A Practical Introduction*. Oxford. Oxford University Press, 1998. ISBN 9780198700166, 0198700164.
- GROSZ, B. J. y SIDNER, C. L. Attention, intentions, and the structure of discourse. vol. 12, páginas 175–204. 1986.
- HARPER, S. Night of the living dead: Reappraising an undead classic. En *Bright Lights Film Journal*, 50. 2005.
- JARVIS, J. y BERENT, I. Storycraft. En <http://www.writerspage.com/>. 2010.
- KASPAROV, G. K. y KING, D. *Kasparov against the world : the story of the greatest online challenge*. New York : KasparovChess Online, Inc., 2000. ISBN 0970481306, 9780970481306.
- KIM, J. Events as property exemplifications. En *Events (International Research Library of Philosophy, 15)*. Aldershot: Dartmouth Publishing, 1996.
- KLEIN, S., AESCHLIMANN, J. F., BALSIGER, D. F., CONVERSE, S. L., COURT, C., FORSTER, M., LAO, R., OAKLEY, J. D. y SMITH, J. Automatic novel writing: A status report. 186. Science Department, University of Wisconsin, 1973.
- KWONG, O. Y. Constructing an annotated story corpus: Some observations and issues. En *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)* (editado por N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odiijk, S. Piperidis, M. Rosner y D. Tapias). European Language Resources Association (ELRA), Valletta, Malta, 2010. ISBN 2-9517408-6-7.
- LANG, R. R. *A Formal Model for Simple Narratives*. Ph. d. thesis, Tulane University, 1997.
- LANG, R. R. A declarative model for simple narratives. En *Proceedings of the AAAI Fall Symposium on Narrative Intelligence*, páginas 134–141. AAAI Press, 1999.
- LIN, D. An information-theoretic definition of similarity. En *ICML*, páginas 296–304. 1998.

- LOVECRAFT, H. P. *Supernatural Horror in Literature*. Courier Dover Publications, 1973. ISBN 0486201058.
- LOVECRAFT, H. P., JOSHI, S. T. y CANNON, P. *Herbert West-Reanimator in More Annotated Lovecraft (1st ed.)*. New York City, NY: Dell, 1999 (1922). ISBN 0-440-50875-4.
- LU, R. y ZHANG, S. Automatic generation of computer animation: Using ai for movie animation. vol. 2160. Springer Verlag, Berlin-Heidelberg-New York, 2002.
- LYTINEN, S. L. Conceptual dependency and its descendants. *Computers & Mathematics with Applications*, vol. 23(2-5), páginas 51 – 73, 1992. ISSN 0898-1221.
- MEDLER, B. y MAGERKO, B. Scribe: A tool for authoring event driven interactive drama. En *TIDSE*, páginas 139–150. 2006.
- MEEHAN, J. R. Tale-spin and micro tale-spin. En *Inside computer understanding*. Lawrence Erlbaum Associates, Hillsdale. Roger C Schank and Christopher K Riesbeck, editors, 1981.
- MONFORT, N. *Generating Narrative Variation in Interactive Fiction*. Ph. d. thesis, University of Pennsylvania, 2007.
- O'REILLY, T. What is web 2.0. *Design patterns and business models for the next generation of software*. En <http://oreilly.com/web2/archive/what-is-web-20.html>. O'Reilly Media, Inc., 2005.
- PEINADO, F. *Un Armazón para el Desarrollo de Aplicaciones de Narración Automática basado en Componentes Ontológicos Reutilizables*. Ph.d. thesis, Universidad Complutense de Madrid, 2008.
- PEINADO, F. y GERVÁS, P. Minstrel reloaded: From the magic of lisp to the formal semantics of owl. En *TIDSE*, páginas 93–97. 2006.
- PIZZI, D. y CAVAZZA, M. From debugging to authoring: Adapting productivity tools to narrative content description. En *ICIDS*, páginas 285–296. 2008.
- PORTEOUS, J. y CAVAZZA, M. Controlling narrative generation with planning trajectories: The role of constraints. En *ICIDS*, páginas 234–245. 2009.
- RUSSO, J. *Return of the Living Dead*. Dale Books, 1978. ISBN 0-89559-062-X.

- SCHILDER, F. *Representation and Management of Narrative Information: Theoretical Principles and Implementation - Gian Piero Zarri*, Springer Verlag, 2009, x+301 pp; ISBN 978-1-84800-077-3, vol. 36. 2010.
- SUROWIECKI, J. *The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations*. Doubleday, 2004. ISBN 0385503865.
- SWANSON, R. y GORDON, A. S. Say anything: A massively collaborative open domain story writing companion. En *ICIDS*, páginas 32–40. 2008.
- TODOROV, T. *Gramática del Decamerón*. 1973. ISBN 8473300041.
- TURNER, S. R. Minstrel: A computer model of creativity and storytelling. En *Technical Report UCLA-AI-92-04*. Computer Science Department, University of California, 1992.
- WIERZBICKA, A. *Semantic Primitives*. Frankfurt: Athenäum, 1972. ISBN 376104822X.
- WIKINOVELA. En <http://www.wikinovela.org/>. Universidad de Deusto, 2006.
- ZADEH, L. A. Fuzzy logic = computing with words. En *Computing with Words in Information/Intelligent Systems 1*. Springer, Valletta, Malta, 1999. ISBN 978-3-7908-1217-6.