Automated Planning of Children's Stories using Causal Links, Agents and Commonsense Ontology

Ethel Chua Joy Ong[1] and Jasmine Irene Ng[2]
College of Computer Studies, De La Salle University
Manila, 1004 Philippines
ethel.ong@delasalle.ph[1], jasmine_ng@dlsu.ph[2]

ABSTRACT

Building computer systems that have the ability to generate human understandable stories has been the subject of research in the field of computational narrative, as these can find applications in education and entertainment. But story planning, which involves the identification of a story plot structure and the production of a coherent and interesting sequence of events that lead story characters to perform actions and experience emotions in order to achieve their goals, remain a challenging task.

In this paper, we present a comparative discussion and analysis of the approaches employed by Picture Books 2 (PB2) and its variants, PB2 Planning Agents (PB2-PA) and PB2 ConceptNet (PB2-CN), to produce children's stories of the fable form. The comparison is aimed at determining whether the enhancement of the storytelling knowledge and the use of agents that can reason over this knowledge during story planning would lead to the generation of stories that contain more cohesive story events. Specifically, two forms of enhancement were made; first, by using pre- and post-conditions as additional criteria in the selection of candidate events; and second, by using existing language resources to supply the necessary commonsense knowledge needed by the planner.

Comparative evaluation of the three systems using the criteria on *coherence and cohesion*, *story elements* and *content* showed that PB2-PA received the highest overall average score of 4.25 out of 5. This is followed by PB2 which garnered an average score of 3.81. PB2-CN garnered the lowest average score of 3.29. The use of three agents (character, plot and world agents) in PB2-PA led to the generation of better story plans containing character actions that are more consistent and directed to the selected theme. The use of existing resources in PB2-CN, though appropriate and relevant to the identified story themes, is found to be insufficient to support the storytelling knowledge requirements of the planner.

Keywords

Story Generation, Story Planning, Agents, Commonsense Knowledge

1. INTRODUCTION

Continuous efforts have been done at the Center for Language Technologies[1] to build intelligent computer systems that can mimic the human art of narrating children's stories. Dubbed as the "Picture Books" story generation systems, each variant produced its own "kind" of stories, particularly fables, for children age 4 to 8 years old. Though the original intent of the research has been to address the computational modeling of storytelling knowledge and the automated planning of the narrative flow, recent efforts have also geared towards designing highly interactive story-based environment that can teach young children important values such as honesty and friendship while building their vocabulary skills. A common story pattern is evident in the stories that are generated, where the main child character violates the lesson, suffers from the consequences of his/her action/s, and eventually learns the intended lesson.

The first story generator in the series, Picture Books 1 [9], is targeted for children age 4 to 6 years old. A Picture Editor facility lets the child choose a background and place character and object stickers to the selected background to form a picture that will serve as the input to the story planner. Then, the system generates a story text based from this input picture.

Even though Picture Books 1 received surprisingly positive results based from the experts' evaluation and shows a lot of potential in the story generation field, there are problems with the stories being generated. Since the system limits the children by only allowing single-scene input pictures, the generated stories may not necessarily match the original intent of the child as he/she selects the story's background and its characters and objects.

Picture Books 2 [2], the second story generator in the Picture Books series, tries to address this issue. Targeted for children age 6 to 8 years old, its Story Editor facility allows children to define multiple scenes with the same background to comprise a picture story that serves as input to the system. By doing so, children can visually specify character and object movement and appearance/disappearance across two adjacent scenes while enabling the system to output more complex stories that contain character actions and events highlighting these transitions.

Picture Books 2 (PB2) stories are also set in more adventurous places such as the *camp* and the *street* to help the older children learn to explore the world on their own. Furthermore, the story characters have each been given three positive traits and three negative traits to help children relate to these characters better. These traits and the transitions between scenes are then used as additional factors in planning the story to be generated.

In this paper, PB2 and its variants – the PB2 Planning Agents [3] and the PB2 ConceptNet [21] – will be the main focus of discussion. The next section begins by providing the motivation for pursing the PB2 Planning Agents (PB2-PA) and the PB2

[1] A research laboratory at De La Salle University - Manila that pursues research works dealing with the processing of human languages.

Philippine Computing Journal Dedicated Issue on Natural Language Processing, pages 1–10
Vol. IX No. 1 August 2014

ConceptNet (PB2-CN) research. This is followed by a detailed discussion of the approaches in story planning employed by each of these systems. Section 3 presents a comparative analysis of the results from evaluating the three systems. The paper ends with a summary of our findings and recommendations for further work to continuously enhance the quality of the generated stories.

2. STORY PLANNING IN PB2

The development of PB2 [2] was motivated by three main factors - i) to allow the child to specify his/her input picture containing a sequence of at least three scenes depicting appearance or disappearance, and movement of characters or objects; ii) to increase character believability by giving positive and negative traits to each of the story characters; and iii) to explore the use of causal event planning instead of the pre-defined author goals employed in Picture Books 1 [9]. However, without a reasoning engine with rules for checking the logical consistencies of story events, instances of stories that make no sense in terms of character actions and responses to events were generated.

PB2-PA [3] provided three solutions to address this limitation. First, an enhanced representation of events that include pre-condition and post-condition criteria was implemented. Second, a model representing the current state of the story world is also included to track the physical and emotional states of story characters, as well as the states of the different objects that these characters manipulate or use for interaction as the story progresses. This model is one of the five levels of story knowledge representation defined by Oinonen and his colleagues [12]. The last approach is the utilization of three agents in planning the story, which was adopted from the multi-agent framework of the Virtual Storyteller [19].

At the onset of the Picture Books research, it has been recognized that a collection of knowledge is a necessary pre-requisite for computers to generate stories. The basic design of this knowledge base adapted the binary conceptual relations of ConceptNet [8]. This is a semantic network of commonsense concepts that are classified into thematic categories such as events, causal and affective, which fit with some of the characteristics inherent in stories. It was argued that ConceptNet may not contain the necessary concepts that reflect the daily activities of children, such as *playing*, *going to school*, and *staying safe and healthy*. Thus, each of the Picture Books systems built its own commonsense semantic ontology that has been manually populated with concepts relevant to the set of themes supported by a specific story generation system [13]. *But can ConceptNet provide the necessary knowledge needed by computer story generators?* This was the primary research question that motivated the development of PB2-CN [21].

The next three sections provide a detailed discussion of the planning algorithms employed by each of the three PB2 systems.

2.1. Causal Event Story Planning

PB2 employs a theme-based cause-effect planning [2] algorithm to generate a story for a given multi-scene input picture. A sample input picture with three scenes is shown in Figure 1. Its corresponding story text is shown in Listing 1. Note that in generating the story, the planner must take into consideration the location and placement of characters and objects in each of the scenes. Character and object relocation or movement must be identified and a corresponding story text describing these movements (referred to as 'scene transitions') must be generated.

Figure 1. Sample Input Picture and Story

Listing 1. Sample story from Picture Books 2

Title: Helen the Hippo learns to be Brave

It was a fine evening. Helen the hippo was in the camp for a trip. She brought a packed marshmallow.

The camp is very far. She felt hungry. …

Helen the hippo saw a shadow. She felt scared. She turned on a flashlight. …

Helen the hippo learned that when she is scared, she should search shadow. From then on, Helen the hippo learned to be brave.

To accomplish this, the story planner of PB2 has three subcomponents, namely theme formulator, setting formulator and event generator. Three sources of knowledge – the semantic ontology containing commonsense assertions, the character traits model, and the attributes of the input stickers (i.e., characters and objects in the story world) – are also made available to support the planning task. These components and their interactions are depicted in Figure 2.

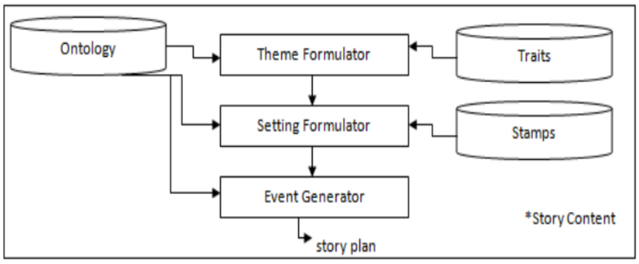

Figure 2. Story Planner of Picture Books 2 [1]

Given an abstract story representation of the input picture from the story editor, the theme formulator uses the *CausesConflictOf* relations in the semantic ontology to identify the set of candidate conflicts for the story. The selected conflict is then used as the basis for the story theme. For example, if a story character is not brave, the conflict may revolve around story events highlighting his/her being afraid or scared of something, as shown in Table 1. The story theme can then be about *learning to be brave* (or to overcome one's fear), as narrated in the sample story in Listing 1.

Table 1. *CausesConflictOf* relations between non-Character traits and *Conflict* concepts

Concept 1	Relation	Concept2
Brave	CausesConflictOf	Scared
Responsible	CausesConflictOf	Lose
Obedient	CausesConflictOf	Disobey

Figure 3. Story Planner of Picture Books 2 [2]

The setting formulator uses the theme and the selected background to determine the time when the story takes place. For example, using the *learning to be brave* theme and a *camp* background, the time can be set to *evening*.

The event generator takes the adjective used to describe the background as a starting point to find a path in the semantic ontology that will lead to the identified conflict. For example, the *camp* can be described with the adjectives *far* and *crowded*. From one of these adjectives as the starting node, the planner follows a chain of *EffectOf* relations to reach the target node containing the conflict concept. Figure 3 illustrates a sample path depicting the cause-effect link between "story events" concepts, starting from the background adjective *crowded*, leading to feeling *dizzy*, necessitating a *sleep* action, until finally the target node containing the conflict concept, e.g., *scared*, is reached.

To increase the length of the story, the event generator also checks for candidate sub-events to support the description of the main events. These sub-events are depicted in Figure 2 as orange-colored nodes, for example, to go to *sleep*, the character may need to *take a bath*, *brush* his/her teeth, *comb* his/her hair, or *pray*.

The process is repeated in order to find a path from the conflict, i.e., *scared*, to a possible resolution, e.g., the character should *search* for the cause (*what is making the scary sound in the night?*). The *HasResolution* relations in the semantic ontology, shown in Table 2, are used at this point in the planning process.

Table 2. *HasResolution* relations between *Conflict* concepts and *Resolution* concepts

Concept 1	Relation	Concept2
Scared	HasResolution	Search
Lose	HasResolution	Admit
Disobey	HasResolution	Apologize

Variances in the generated story (for a given background, character trait, and objects) are achieved by populating the semantic ontology with sufficient conceptual relations to allow the planner to randomly select a candidate node to pursue.

2.2. Agents in Story Planning

PB2 selects a path among several candidate chains of events on a random basis. This can, at times, lead to the production of illogical stories that contain unbelievable character actions, as shown in the short story text in Listing 2. The underlined story events about Peter "*helping his mother*" and "*preparing to go home*" are not related to the rest of the story which is about Peter "*playing in the park*".

Listing 2. Sample story from PB2 with unbelievable actions or events

Title: Peter the Pig learns to be Helpful

One bright afternoon, Peter the pig was in the park for relax. He got a frisbee.

The park was very clean. He helped his mother. He prepared go home.

Peter the pig brought the frisbee. He played the frisbee park. He knocked out his food. He felt lazy.

Peter the pig did not know what to do. He felt guilty. He helped. He was not lazy anymore.

PB2-PA [3] tried to address this problem by improving on the reasoning engine of the story planner. Instead of using only the *EffectOf* relations, a more detailed event (and action) model has been designed to include pre-conditions and post-conditions that were used as criteria for the selection of candidate events or character actions.

2.2.1. Event (Action) Model

Table 3 contains the structural representation for an event. The model includes the specification of the required event parameters, namely `#agent` and `#patient` or the doer and receiver of the event, respectively, and an optional parameter `*location` representing the location where the event can take place.

The model also contains variables (preceded by the symbol '?') that are used to refer to a state or element in the world. States include the following:

- `?know` and `?holds` to represent what the character currently knows or is holding, respectively;
- `?hasProperty` to represent the current property of an object; and
- the world element such as `?mainchar` to refer to the main story character

Table 3. Structure of an Event [3]

Category	event
Concept	*lose(*#agent, #patient, *location)
Primitive	GRASP
Pre-condition	`?inCareOf(#agent, #patient) ∧ ?know(#agent,?location(#patient)) ∧ ?isA(#agent, character) ∧ ?isA(#patient, object)`
Post-condition	`¬?holds(#agent, #patient) ∧ ¬?know(#agent,?location(#patient) ∧ ?hasProperty(#patient, lost))`

Primitives, adapted from [17], are the most basic action units, such as `ATRANS` (transfer of abstract relationship, i.e., *give*) and `PTRANS` (transfer of physical location of object, i.e., *grasp*). These basic action units form part of the transition types supported by PB-PA. For example, the concept *lose* has the transition type `DM_PTRAN`, where *D* indicates disappearance and *M* indicates movement, as shown in Table 4.

Table 4. Transition Types

Transition	Description
D	Disappearance (of a story element)
A	Appearance
M	Movement
N	No Transition

The pre-conditions specify constraints that must be satisfied before an event can be executed. In Table 3, the pre-condition for event "*lose*" states that the character (which is the *agent*) has the object (which is the *patient*), thereby also stating that the character knows the location of the object.

Post-conditions are the resulting states that the story world must hold after the event is executed. For the event "*lose*", its post-condition states that the character is not holding the object and does not know the location of the object. The object has the property *lost*, which is defined, in this case, as the absence of the object from the scene in the input picture.

Actions are events that are explicitly performed by a story character, as opposed to naturally occurring events or events that occur as a result of some other events or actions. An action follows the same structural representation as an event.

2.2.2. World State Model

A story world state model is used to track the changing physical, spatial and emotional states of characters and objects as a result of executing events or performing character actions. The model has been derived from the Virtual Storyteller [19] that modified Trabasso's original General Transition Network (GTN) [20] to comprise story elements and causal relationships for use in story generation.

The PB-PA planner uses the story world state model to determine if the pre-conditions of an event have been satisfied, thus allowing the event to take place in the story world. The story world model is also updated after the completion of an event, and the new values are determined from the post-conditions associated with the completed event.

The story world state model contains the following dynamic attributes for each character in the story – the social event the character is currently in, its emotion towards another character, its feelings, its perception regarding the world, its physical states and bodily needs. These attributes are shown in Table 5.

Table 5. Story World State Model for a Character

Bodily Needs	Food	False
	Water	False
	Rest	False
	Excitement	False
	Comfort	False
Physical State	Mind State	Normal
	In Care Of	Water Jug
	Holds	Water Jug
Emotion	Love(Peggy)	
Feelings	Null	
Perception	Hear() Smell() Know(?location(waterjug, camp)) Feel() See() Guess() Believe() Taste()	
Social Event	Null	

A world map is used to represent the story world state model. It tracks the location of the elements (character or object) in the 36-grid background, as shown in Table 6. The division of the background into 36 grids is a design decision that has been adopted from PB2 [1]. This guides the planner in identifying the distance of story elements from each other.

Table 6. World Map Representation

Grid #	Element	Type
3	Water Jug	OBJECT >> Concept Noun
36	Peter	CHAR >> Concept Noun

2.2.3. Character, World and Plot Agents

PB2-PA utilized three agents, namely the character agents, the world agent and the plot agent, as shown in Figure 4. The character agent takes care of generating the candidate actions of a character given his/her goal. It ensures that the character's action is justified by his/her predefined trait, belief and desire in order to achieve character believability as described in [15]. A character frame is maintained for this purpose and contains predefined information such as the character's name, gender, traits, roles and desires, as shown in Table 7.

Table 7. Static Model of a Character

Name	Peter	Main Character
Traits	Negative	Responsible, Helpful, Obedient
	Positive	Honest, Brave, Persevering
Roles	Sister(Peggy)	
Desires	Play, Sweets	

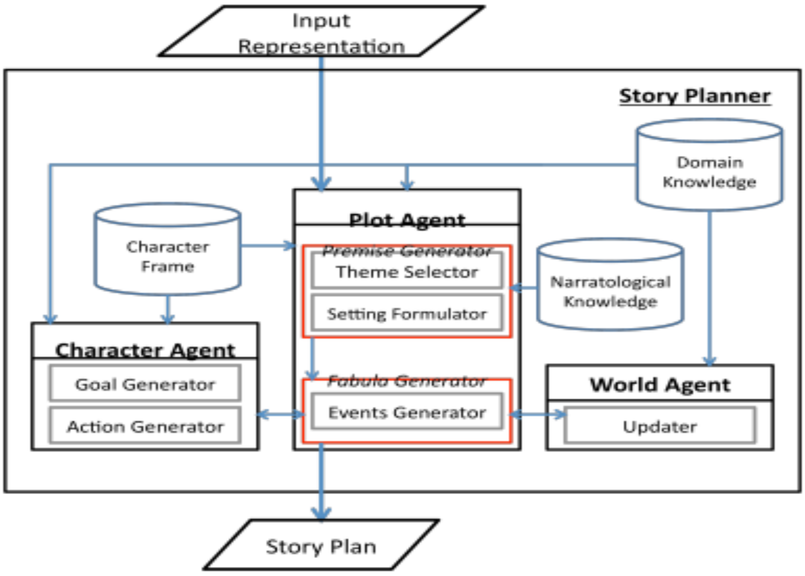

Figure 4. PB2 Planning Agents Architecture [3]

Because a character may have multiple goals, the first task of the character agent is to prioritize these goals based on their intensity, with goals triggered by emotions being given the highest priority, followed by goals toward the satisfaction of a bodily need, and finally by the desires. The character agent evaluates its goal after every execution of an event by checking the world state model.

Once a goal has been identified, the character agent generates the plan of action for the character to accomplish its goal. This plan of action contains a causal chain of events that is retrieved from the domain knowledge (semantic ontology). An example plan of action that is generated for the goal "*find*" is as follows, which states that to find "*an object*", a character has to perform a "*search*" action that, if successful, will lead to the "find*ing of the target object*" event:

```
Goal(find) := action(search) ^ event(find)
```

An event is not explicitly included as part of a character's plan; it occurs as a natural phenomena or as a result of some character action. However, given that the event is part of the condition for the goal to be considered “accomplished”, it is included as part of an action plan. A goal is considered “accomplished” if the post-conditions defined for that goal have been satisfied.

The world agent manages the story world state model and knows the state of every character and every object present in the story at a particular point in time. It can also determine if an event can take place in the story world by checking whether or not the current world state model has satisfied the pre-conditions of an event. After the execution of an event, the world state is updated again by applying the post-conditions of the event. The same validation and update processes are applied for the execution of a character’s action.

The plot agent acts as the overall story director. It is responsible in ensuring that the selected story path will lead to the identified theme by approving or disapproving character actions. The best-fit theme is selected based on the negative traits of the main character, and by the presence and positioning of the characters and objects in the input picture. Every condition that is satisfied by the input picture is given a score (a single point). The candidate theme with the highest score is then chosen.

The setting for the story is then formulated for the best-fit theme based on the desires of the main character and the set of events that can be executed in the user-selected story location. Additional support characters, their relationships and empathy with the main character, and the objects that are in possession of the main character may be generated as needed to produce a story plan.

The plot agent uses a collection of narratological knowledge (described in detail in [3]) comprising of themes and author goals to drive the story flow. The author goals, first introduced in Picture Books 1 [9], were modified to work with the agent-based planner. It contains the three essential parts of a story, which are the character and his/her environment, the conflict and its counter action, and the conclusion or the final state of the world.

A mechanism employed by the story planner to introduce the character and his/her environment is with the use of the character's desires. If the character’s desire can be executed in the input location, then the initial goal of the character is immediately set to perform an action in that location.

Whereas PB2 relies on the events generator to derive the chain of events, the plot agent of PB2-PA works with its character agent to verify if the action proposed by the latter could lead the story character to the conflict, then to experience its consequences and finally to the resolution of the conflict. In cases of deviations, the plot agent formulates possible events to force the character to change his/her goal, thus triggering a change in his/her plan of actions towards the achievement of the story theme. This may include, among others, the performance of an action that necessitates the explicit use of objects and the possible introduction of new characters into the story. If no story can be generated, the plot agent selects the next highest scoring theme and performs the same process again.

Listing 3 shows a sample story with the theme "*learning to be responsible*" using the plot "*lose-search-find*". Below is the list of the other types of story text that PB2-PA is able to generate because of its expanded models of story world elements:

a) The introduction of the main character (*Peter*)'s empathy towards a secondary character (his sister, *Peggy*) derived from the main character's "*emotion*" attribute in Table 5;

b) The description of the character's bodily need (*Peter felt thirsty*); and

c) The character's perception about the placement of objects in the story world (*Peter realized that he lost Peggy's water jug*).

Listing 3. Sample story generated by PB2-PA

> Title: Peter learns to be responsible
>
> One evening, Peter wanted to play in the camp. Peter loves his sister, Peggy. Peter borrowed Peggy's water jug from Peggy.
>
> Peter went to the camp. Peter played in the camp. Peter lost Peggy's water jug in the camp.
>
> Peter felt tired. Peter felt thirsty thus Peter wanted to drink water from the water jug. Peter realized that he lost Peggy's water jug in the camp. Peter felt scared.
>
> Peter searched Peggy's water jug in the camp because Peter wanted to find Peggy's water jug. Peter found Peggy's water jug in the camp. From then on, Peter learned to be responsible.

2.3. ConceptNet in Story Planning

One of the primary reasons why the task of making computer systems generate logical stories is difficult is because of their lack of a sufficient collection of commonsense knowledge that humans inherently possess in order to understand and create stories. Previous systems made use of knowledge bases that have been manually populated with concepts specifically for a particular domain dictated by the set of story themes to be generated; thus, they will not work for inputs that are not expected by these systems. PB2-CN [21] tried to address this constraint on the knowledge base by exploring the use of existing large commonsense knowledge and linguistic repositories, specifically ConceptNet [8], VerbNet [10] and WordNet [11].

Because the available resources, despite their volume, are still insufficient to support the tasks of the planner, PB2-CN used a two-layer ontology that was adapted from Swartjes [18] to represent storytelling knowledge. The Upper Story World Ontology (SWO) contains information pooled from the three existing commonsense resources stated above, with the knowledge representation following the binary semantic relations of ConceptNet. The Domain Specific SWO contains knowledge needed to be able to generate a story such as characters, roles, traits, author goals and vocabulary.

2.3.1. Upper Story World Ontology

The upper story world ontology contains assertions that were extracted semi-automatically from ConceptNet, such as the first two assertions in Table 8. Assertions containing concepts that may not be appropriate for young children, such as those dealing with violence, death, alcoholism, gambling and other sensitive information were discarded. Assertions containing concepts that may be important for telling stories but are, however, found to be lacking in ConceptNet, were extracted from other sources. Specifically, these assertions include those that deal with facts concerning the transfer of possession, shown in the last 4 rows of Table 8. These were extracted from VerbNet (*steal* and *buy*) and WordNet (*kidnap* and *acquire*).

Table 8. Assertions from ConceptNet, VerbNet & WordNet

Source	Relation	Target	Score
child	desires	good grade	3
take exam	causes	pass course	2
steal	causes	has_possession	2
buy	causes	has_possession	2
kidnap	causes	has_possession	2
acquire	causes	has_possession	2

Out of the 15 categories of synsets for verbs in WordNet, only those assertions with a supersense value of "*verb.possession*" were extracted as this supersense refers to verbs that denote a transfer of possession. Similarly, VerbNet classes that contain the "*has_possession*" predicate were also added to the ontology.

For assertions extracted from ConceptNet, the scores were copied as is. For VerbNet and WordNet, a default score of 2 is set since this is the ideal threshold value for an assertion to be considered reliable by the story planner.

Other assertions were also added manually to fill-in gaps that are necessary to generate stories with target themes, e.g., honesty during an exam. These are shown in Table 9.

Table 9. Assertions that were manually added

Source	Relation	Target	Score
cheat exam	causes	pass course	2
cheat exam	causes	fail exam	2
steal	causes	scold	2
steal	causes	fight	2

2.3.2. Event Model

Table 10 contains the structural representation of an event data. Each event data refers to the event that happens in the story and represents a single sentence in the output. There are four types of events that are currently supported, namely, description, experience, action and goal.

Table 10. Structure of an Event with 3 example Event Types

Attribute	Action Event	Specific Desire	General Desire
Name	pass course	has_possession	learn
Type	action	goal	goal
Score	3.87	3	5
Constructs			learn.v.01 learn-14
Trait	lazy		
Story Roles	Agent: *Danny*; Patient: *course*	Agent: *Danny*; Patient: *toy*	Agent: *Danny*
Polarity	1	1	1
Goal status	-NA-	0	0

Description events are used to describe aspects of the world, such as a character's trait or role. The required parameters are the name that stores the concept being described (e.g., the main character), the event's polarity, and a "predicate" attribute to hold the descriptive concept (e.g., the specific trait or role). Experience events are used to describe a sequence of events. The required parameters are the name and a list of events.

Action events (second column in Table 10) are the voluntary actions performed by a character. All attributes for an event are included, with an additional "verb" attribute that indicates the action that has occurred. Lastly, goal events represent the desire or want of the character. This can either be a goal of wanting to have possession of an object (third column of Table 10) or a general goal or desire (fourth column of Table 10).

The event should have a score value of at least 2 for it to be considered by the story planner. Moreover, stored in the "constructs" attribute are the WordNet and VerbNet senses of the event. The character traits associated with the events are also represented. The story roles indicate the set of thematic roles that have been adapted from VerbNet, and these are agent, patient, location, instrument and source. As for the polarity of the event, a value of 1 indicates a positive event, 0 is a negative event but not deliberately performed on purpose by a character, and -1 is also a negative event and deliberately performed on purpose. The goal status attribute is used for goal events only to signify whether the goal is being pursued (0) or if it has already been achieved (1) or if the character was unsuccessful in achieving the goal (-1).

2.3.3. Writing the Story Plan

Story planning begins with a prompt (shown in Figure 5) that asks the user to select story elements such as the background, characters and objects to be included in the story. The characters should be chosen from a pre-defined set of available story characters while the objects and background can be anything the user wants.

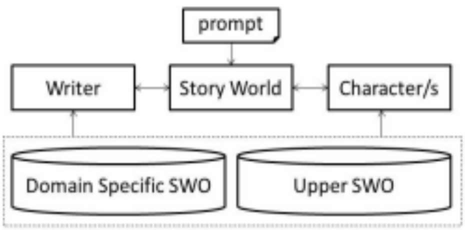

Figure 5. Picture Books 2 ConceptNet Architecture [21]

The writer then randomly selects at least one main character from the user-provided prompt. Since the story follows a pattern where the main character performs a behavior that is associated with his/her negative trait and progresses until he/she learns the lesson, the writer has to check the story world at every timepoint to see if the story theme has been achieved.

The main characters formulate their own goals by querying Desires(?role, goal) from the upper story world ontology. For each role of a main character, there is a corresponding set of candidate goals that is retrieved from the. The type of the goal depends on the character's desire. It can be a goal with a desire to *possess an object*, or a goal *to achieve some event or action*, *such as learn and play*. Each desire is given a score that signifies how valid or acceptable the candidate assertion is. Only those desires that have scores of at least 2.0 can be used by the planner. Furthermore, the higher the score, the higher are the chances of the corresponding desire to be selected. Consider the following query sent by the planner to the ontology:

"Desires('child', goal)"

The ontology may return the following available assertions:

Desires(child, learn) +2
Desires(child, play) +5

In this case, "Desires(child, play)" would be selected.

The main characters should also devise a set of actions that they should do in order to achieve their respective goals. These candidate actions are again queried from either the domain-specific ontology or the upper story world ontology. For an action coming from the domain-specific ontology to be selected, its pre-conditions must be satisfied by previous events that have already taken place in the story world, and its post-condition should be equivalent with the goal.

Consider the action structure shown in Table 11. A character who has a negative trait of being dishonest might achieve his goal of having a good grade by performing the action "*cheat*" if the action's pre-condition ("*dishonest*") and post-condition ("*good grade*") match the trait and desire, respectively, of the character.

Table 11. Structure of an Action "*Cheat*"

Attribute	Description
Name	Cheat
Type	Action
Score	
Constructs	
Trait	Dishonest
Story Roles	Agent: *Danny*; Patient: *exam*
Polarity	-1
Pre-conditions	Dishonest
Post-conditions	Good grade

On the other hand, a story graph is created by expanding every unexplored node if the character agent queries the upper story world ontology for its candidate actions using the following:

MotivatedByGoal(?event, goal),
HasPrerequisite(goal, ?event),
Causes(?event, goal),
UsedFor(?event, goal).

From the resulting story graph, the character searches for candidate paths that have a length of three (or three nodes away from the goal node). This optimal value for the path length was derived from an empirical test conducted which suggests that paths with length 4 or higher produces unreliable story events.

The writer's responsibility is to ensure that the path gathered by the characters adhere to the intended plot structure and if the user input satisfies these actions. The writer will only retain paths that contain an action related to the negative trait of the character, lead to a consequence and eventually reach the goal. This is necessary to achieve character believability, where character actions are dictated by their traits. The story elements such as characters, objects and locations needed by the events in the candidate paths are also checked if they are present in the prompt. The writer then forwards the partial story plan containing the validated paths to the story world.

2.3.4. Story World

The story world model in PB2-CN contains a history of events that have occurred so far in the story. The execution of each event found in the partial story plan changes the state of the story world. This state is referred to as a timepoint. Both the writer and the characters execute the partial story plan by interacting with the story world. They interact through querying from the storytelling knowledge and passing the results to the story world. The writer is given a full access to all the information in the two-level ontology while the characters can only access certain information.

For each event in the story plan, there are three basic rules followed in order to assert events into the story world. First, if the character is currently not pursuing any goal, then the character should pursue its goal and this is asserted to the story world. If the goal has a status of failed (value of -1), the failed goal together with a new description event is asserted to the story world. This description event is used to generate a story text that will describe the character's trait that led to the failed event, e.g., *Danny did not study hard because he is lazy.* This is depicted in Figure 6, where the description event on the right is created to provide an explanation for why the action on the left was deliberately not performed by the character (polarity = -1).

Action	
Name	study hard
Type	action
Verb	read
Score	3.7
Constructs	...
Traits	lazy
Story roles	Agent: Danny
Polarity	-1

→

Description	
Name	Danny
Predicate	lazy
Polarity	1

Figure 6. Action event and resulting Description event

Second, if the character is the agent of the action, it executes the action by passing the event to the story world. Otherwise, if it is at least involved in the event, it should react accordingly. Finally, all other events classified under experience are asserted as is to the story world.

A sample story plan is shown in Listing 4. The events in the story have been formatted in English for easier reading, though the system currently does not produce any surface form of the text.

Listing 4. Sample story plan from PB2-ConceptNet

Input: Danny (lazy)

[1] Danny is a student. Danny wants good grade.
[2] Danny is lazy. Danny not take note using pocket size notebook. Danny sleep.
[3] Danny cannot pass course.
[4] Danny does not good grade.
[5] Danny realizes mistake of not take note because of lazy.

A task of the writer is to assert the necessary description events into the story world in order to give context to the events that happen in the story. For example, when a main character asserts a goal event into the story world for the first time, the writer must then assert a description event describing the role of the character in order to rationalize the formulation of the goal. In Listing 4, the text "*Danny is a student.*" corresponds to the description event that describes the role of the main character and explains the formulation of the goal "*Danny wants good grade.*"

3. TEST RESULTS AND FINDINGS

Riedl and Young [16] pointed out that the evaluation of stories generated by an automated story generation system "often relies on subjective assessment". They further noted that subjective assessment can be affected by many factors, including the quality of the language used in the story text, the interest of the reader in the topic of the story, and the novelty of the story plot. They also recommended that objective evaluation be performed using metrics such as story length and assumptions regarding narrative factors that correlate to the generation of better stories. That is, for their research, the narrative factor is the complexity of character roles, which can be measured as "the degree to which the reader perceives character intentionality".

In the case of the PB2 systems, the enhancements were intended to produce stories with a more cohesive sequence of events and character actions that are aligned with the individual character's embodied traits and desires. Furthermore, the stories should remain appropriate for the same target age group, or children age 6 to 9 years old. Thus, the same set of criteria was used for comparing the output of the three PB2 systems. These are *coherence and cohesion*, *story elements*, and *content*.

The coherence and cohesion criterion evaluates the transition between the events and the flow of the story to determine if the generated story makes sense and can be easily understood by the target users. On the other hand, the story elements criterion checks the appropriateness of the character, objects and backgrounds used in the stories. Lastly, the content criterion deals with the appropriateness of the story to the target age group, the sufficiency of the details provided, and the believability of the story events (or the actions performed by the characters).

A fourth criterion, language, which measures the correctness of the sentence structure and the appropriateness of the vocabulary, is not included because PB2-CN did not produce a surface form of the story text as the study focused primarily on the sufficiency of existing linguistic resources in aiding the story planning process.

All versions of Picture Books were evaluated manually by experts in the field of linguistics, child education and story writing in terms of their linguistic and narrative aspects. This approach was retained in evaluating the PB2 systems. The experts provided a score ranging from 1 to 5, with 5 being the highest for each criterion. A total of 10 stories each from PB2 and PB2-PA, and 15 for PB2-CN were evaluated.

For PB2, two sets of stories were evaluated. After conducting the first evaluation, revisions were made to the planner to address some of the issues mentioned by the experts. The stories produced after this revision were the ones compared with the output of PB2-PA and PB2-CN. Table 12 shows a comparative summary of the evaluations on these three systems.

All criteria have received an average score of 3.0 and above, which means that the systems performed generally well. The results suggest that the overall flow of events in the generated stories is generally correct and acceptable. It also means that character believability has been achieved. Furthermore, story elements such as objects and locations are usually used in their proper context and are consistent with the user input.

Table 12. Evaluation Results for the three (3) PB2 Systems

Criteria	PB2	PB2-PA	PB2-CN
Coherence and Cohesion	3.66	4.25	3.33
Elements	4.02	4.30	3.26
Content	3.76	4.20	3.28
Average	**3.81**	**4.25**	**3.29**

Because there is no consistency in the individual scores of the stories generated by these systems, it cannot be generalized that the stories are good or not. In some cases, the systems can produce stories that receive high average scores in all criteria but in other cases, low scores were gathered. High-scoring stories have smooth flow and appropriate transitions of events whereas low-scoring stories provide unbelievable flow of events and sentences that consist of incorrect structure or grammar and cannot be comprehended.

PB2-CN garnered the lowest average scores in all criteria because its storytelling knowledge has been minimized to test the sufficiency and effectiveness of the contents of ConceptNet in supporting the knowledge requirements of the story planner. Thus, even though results showed that ConceptNet is suitable, it cannot be the only source of knowledge for generating stories because there will be cases where information needed by the system is missing and must be gathered from other knowledge base. Because of this, more information has to be added to the upper story world ontology and domain-specific ontology to be able to produce better stories.

On the other hand, PB2-PA garnered the highest average scores in every criterion among the three systems because the planning algorithm gives priority to story plans that adhere to the selected theme. Furthermore, the use of a plot agent assures the actions of the characters are consistent and directed to the selected theme. Overall, the use of the three agents (character agent, plot agent and the world agent) has been proven to be helpful in its story planning tasks.

4. CONCLUSION

In this paper, we have compared the planning processes of Picture Books 2 and its extensions, PB2-PA that uses three agents in the planning process, and PB2-CN that uses available resources to provide the planner with the knowledge it needs to do its task. The extensions were meant to address the limitations of PB2 along two aspects – 1) in planning, by delegating the task of planning the actions to the individual character agents to promote character believability while maintaining a plot agent or a writer to ensure overall adherence of the story events to the identified theme and story plot structure; and 2) in the sufficiency of the knowledge base, by reducing the reliance on manually populated knowledge bases and turning to readily available linguistic resources, namely ConceptNet, WordNet and VerbNet.

Since creating an automatic story generation system that is capable of producing stories that vary greatly from one another is mainly due to how diverse and large its commonsense knowledge and storytelling ontology is, investing in these knowledge bases will be beneficial. One can explore projects like Never-Ending Language Learning (NELL) [6] that crawls over millions of web pages to learn new knowledge to append knowledge to the story generator. However, just as PB2-CN had to use a threshold on the assertions that the planner can use in generating story events, additional works to filter NELL's knowledge base should be done in order to extract only those knowledge that are appropriate for the target age group and the target domain and/or genre of the stories to be generated. Furthermore, to facilitate learning, the scope of the knowledge and the filtering mechanism can be made dynamic such that it adjusts to the cognitive level of the child as he/she grows older. This way, NELL provides the story generator with age-appropriate knowledge suitable to the child and will continue to support his/her reading due to the variances and increasing complexity and scope of the stories being generated.

Another possible future work is to expand the capabilities of the world agent in managing the story world state model to track the sequences of events that have already taken place. This will allow the planner to take the event history into consideration when identifying the next story events or situations. It can, for example, help prevent the planner from doing sudden shift in the story location without putting a closure on the previous events.

Picture Books stories can be made more interesting if multiple characters and multiple locations can be supported to give the target users a wider range of options on the scope and complexity of the story he/she wishes the system to generate. Having multiple characters and multiple locations will require planners to take into account story events that will ensure characters are interacting with each other to attain their goals, and to identify points in the story when location changes can be effected.

Lastly, the use of stories for sharing knowledge and experiences spans different culture and languages. Research efforts geared towards porting the stories of Picture Books to Tagalog and other Philippine languages would be worth exploring. In 2012, a Filipino surface realizer named FilSuRe [14] was developed at the Center for Language Technologies and used in the realization phase of Picture Books 1 to produce story text in Filipino while retaining the original planning engine. This approach focused the research work on the development of a realizer that can be used not only by story generation systems, but also by text generators. Each PB variant will then have to revise its surface realization phase to use the features of FilSuRe or the surface realizers of any target language.

Another approach is to use systems like Linguist's Assistant [5] to develop a computational model of any language. This linguistic tool can be used to document Philippine languages, specifically, the language's lexicon, ontology and semantics. A simple Linguist's Assistant for Tagalog was explored in [7], which was then used to generate short "story-like" sentences in the language. Picture Books can then be redesigned to generate a semantic representation of its stories in a form that can be fed to the Linguist's Assistant, which will then translate this representation to a target language [4].

5. REFERENCES

[1] Ang, K., Antonio, J., Sanchez, D., Yu, S. and Ong, E. 2010. Generating Stories for a Multi-Scene Input Picture. In *Proceedings of the 7th National Natural Language Processing Research Symposium*, 21-26, November 19-20 2010, De La Salle University, Manila.

[2] Ang, K., Yu, S. and Ong, E. 2011. Theme-Based Cause-Effect Planning for Multiple-Scene Story Generation. In *Proceedings of the 2nd International Conference on*

Computational Creativity, 48-53, Mexico City, Mexico, April 27-29 2011.

[3] Ang, K. and Ong, E., 2012. Planning Children's Stories using Agent Models. In D. Richards and B.H. Kang (Eds.), *2012 International Pacific Rim Knowledge Acquisition Workshop* (PKAW 2012), LNAI 7457, 195-208, September 5-6 2012, Malaysia. Springer-Verlag Berlin Heidelberg.

[4] Beale, S. 2011. Using Linguist's Assistant for Language Description and Translation. In *Proceedings of The 5th International Joint Conference on Natural Language Processing (IJCNLP-11)*, Demonstration Session, Chiang Mai, Thailand.

[5] Beale, S. and Allman, T. 2011. Linguist's Assistant: A Resource for Linguists. In *Proceedings of 5th International Joint Conference on Natural Language Processing (IJCNLP-11), The 9th Workshop on Asian Language Resources*, Chiang Mai, Thailand.

[6] Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka Jr., E. and Mitchell, T. 2010. Toward an Architecture for Never-Ending Language Learning. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence* (AAAI-10), 1306-1313, July 2010, Atlanta.

[7] Castilo, M.A., Go, M., Lam, A.J., Syson, O.B., Xu, P., Ong, E. and Beale, S. Building a Simple Linguist's Assistant for Tagalog. In *Proceedings of the 10th National Natural Language Processing Research Symposium,* 25-31, February 20-21 2014, De La Salle University, Manila.

[8] Havasi, C., Speer, R. and Alonso, J. 2007. ConceptNet 3: A Flexible, Multilingual Semantic Network for Common Sense Knowledge. In *Proceedings of Recent Advances in Natural Language Processing,* Borovets, Bulgaria, September 2007.

[9] Hong, A.J., Solis, C., Siy, J.T., Tabirao, E. and Ong, E. 2009. Planning Author and Character Goals for Story Generation. In *Proceedings of the NAACL Human Language Technology 2009 Workshop on Computational Approaches to Linguistic Creativity*, 63-70, Colorado, USA. Morristown, NJ, USA: Association for Computational Linguistics.

[10] Kipper, K.S. 2005. *VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon*. Ph.D. Dissertation, University of Pennsylvania, Philadelphia.

[11] Miller, G.A. 1995. WordNet: A Lexical Database for English. *Communications of the ACM*, 38(11):39-41.

[12] Oinonen, K., Theune, M., Nijholt, A. and Uijlings, J. 2006. Designing a Story Database for Use in Automatic Story Generation. In *Proceedings of the 5th International Conference Entertainment Computing*, 298-301, Cambridge, United Kingdom.

[13] Ong, E. 2014. *Picture Books: Challenges and Opportunities in Automatic Story Generation*. De La Salle University, Manila. (*In print*)

[14] Ong, E., Abella, S., Santos, L. and Tiu, D. 2011. A Simple Surface Realizer for Filipino. In *Proceedings of the 25th Pacific Asia Conference on Language, Information and Computation* (PACLIC 25), 51-59, December 2011, Singapore.

[15] Riedl, M. 2004. *Narrative Generation - Balancing Plot and Character*. Ph.D. Dissertation, Department of Computer Science, North Carolina State University.

[16] Riedl, M. and Young, R.M. 2005. An Objective Character Believability Evaluation Procedure for Multi-Agent Story Generation Systems. In *Proceedings of the International Conference on Intelligent Virtual Agents*, LNCS, 278-291, September 2005, Greece. Springer-Verlag London.

[17] Schank, R. C. 1975. The Primitive Acts of Conceptual Dependency. In *Proceedings of the 1975 Workshop on Theoretical Issues in Natural Language Processing*, 34–37. Stroudsburg, PA, USA: Association for Computational Linguistics.

[18] Swartjes, I. 2006. *The Plot Thickens: Bringing Structure and Meaning into Automated Story Generation*. Master's Thesis, University of Twente, The Netherlands.

[19] Theune, M., Faas, S., Nijholt, A. and Heylen, D. (2003). The Virtual Storyteller – Story Creation by Intelligent Agents. In Göbel, S., Braun, N., Spierling, U., Dechau, J., Diener, H. (Eds.), P*roceedings of Technologies for Interactive Digital Storytelling and Entertainment* (TIDSE 2003), 204–215. Fraunhofer IRB Verlag.

[20] Trabasso, T., Broek, P. Van den and Suh, S. 1989. Logical Necessity and Transitivity of Causal Relations in Stories. *Discourse Processes*, 12(1):1-25.

[21] Yu, S. and Ong, E., 2012. Using Common-Sense Knowledge in Generating Stories. In P. Anthony, M. Ishizuka and D. Lukose (Eds.), *12th Pacific Rim International Conference on Artificial Intelligence* (PRICAI 2012), LNAI 7458, 838-843, September 5-7 2012, Malaysia. Springer-Verlag Berlin Heidelberg.