IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-6, Issue-9, Sept. 2018 E-ISSN: 2347-2693

Procedural Content Generation in Games towards Semantic Web

V. K. Vashistha[1*], S.K. Malik[2]

[1]USIC&T, GGSIP University, New Delhi, India
[2]USIC&T, GGSIP University, New Delhi, India

[*]*Corresponding Author: vkvashistha@gmail.com*

Available online at: www.ijcseonline.org

Accepted: 24/Sept./2018, Published: 30/Sept./2018

***Abstract*—** Procedural Content Generation (PCG) : A significant research area in the domain of digital games, which provides techniques to automatically generate game content such as levels, narratives, landscape, game rules and mechanics, etc. without or with least human effort. In these days, video games are usually backed by web services in order to fetch game content directly from game servers rather than storing everything at client side, for better controllability over the game content. Semantic Web technologies play an important role in World Wide Web (WWW) with the objectives to create and maintain structured Web of Data to make it more machine understandable. Potentially, Semantic Web may contribute to PCG by enhancing its capabilities in terms of computational creativity, better algorithmic efficiency, scalability, interoperability etc. In this paper, first, the role of PCG and Semantic Web in games has been explored. Second, a Semantic Based PCG Framework has been proposed, which combines strength of both the fields and exploit the content of existing knowledge repositories such as DBPedia, WordNet, Freebase, etc. to generate interesting puzzles. Third, proposed framework has been supported by taking the case study of a popular word game Hangman. Finally, emphasis on exploring various concerns is made towards the role of Semantic Web in procedural content generation in games.

***Keywords*—**Semantic Web, Procedural Content Generation in Games, Quiz Games, Educational Games, DBPedia

I. INTRODUCTION

Procedural Content Generation (PCG) is the automatic generation of content for digital games without or with least human input [27]. It consist of various algorithmic techniques to generate variety of game content such as game levels, rules, mechanics, dungeons, vegetation, terrains, narratives, etc. The core intention of PCG is to reduce the efforts of game designer and to assist them in designing games. It is applied to several successful game titles such as *Ms Pacman, Infinite Super Mario Bros, Minecraft, The Legend of Zelda, Final Fantasy, First Person Shooting Games, Role Playing Games, Racing Games,* etc. It has a vast research area, with it's own challenges and limitations [19], some of which are : generic content generators, interfaces and controllability of PCG systems, generating multiple facets of games such as music, theme, characters along with levels, all at once. Most of the games these days are connected through online services, where game content can be fetched directly from game servers rather than storing everything at client side. This help in better control over game content according to player's preference, demographics, culture, memory constraints, hardware limitations, interoperability, etc. The idea is that these games are becoming web-centric, where semantics and ontology may also play a significant role.

Semantic Web Technologies offer data to be stored structurally in order to make it machine understandable [1][2]. The current web is arranged in the form of hypertext documents which are linked together. These documents have unstructured or semi-structured data in the form of plain text/multimedia which cannot be identified as individual entity. Hence, a user has to explore a lot of documents by querying search engine multiple times with several keywords in order to get the desired result. For example, the query "Whether Person A knows Person B" may require to search and visit multiple documents before getting final conclusion, unless some document explicitly describes relationship between Person A and B. This problem is addressed using *Resource Description Framework (RDF)*, a framework which stores data in the form of triplets <*subject, predicate, object*>. Uniform Resource identifier (URI) is one of the fundamental component of Semantic web Stack, which uniquely identifies an entity throughout the web or within a specific domain. These entities are interlinked together to form a graph, which enables easy information retrieval. The semantic web is an ongoing research world wide having several challenges and limitations like manual insertion and maintenance of consistency, ontology matching, etc. as a key concerns.

Fig. 1: Android Version Hangman Screenshot showing various game components [3] , that is *word* to be guessed, *hint* (here for example "commitment or devotion") and *tag* (here for example "Dictionary")

Hangman is a word game, which serves as source of both education and entertainment to its users. Fig. 1 shows an Android version of Hangman. In this game, player has to guess a word letter by letter, based on given clues, within certain limit of chances. One of the main challenges of developing this game is to determine how the quiz puzzles that a player will be presented are to be gathered. The quiz may cover a broad range of interests. Hence, to manually generate it's content, the designer has to put in a lot of time and effort, exploring the web to create puzzles. The existing PCG techniques greatly rely on some *encoded content representation* of the game configuration [37], which gets computationally evolved to generate new content. However, for quiz games, additional effort is required to collect the actual content itself. Unlike other games, where a lot of permutations of a game configuration can be generated by simply evolving the encoded content representation, in quiz games, content is very close to real world data. It has already been proposed in [3] that how *Search Based Strategy* [37] may be applied to deliver the puzzles in such an order that the player finds the game interesting, but nothing has been discussed about how actual puzzles can be generated. In this paper, extending this work forward, a semantic based PCG framework has been proposed, which leverage the power of Semantic Web technologies by extracting puzzles from open structured data such as DBPedia[1], BBC[2], WordNet[3], etc.

Rest of the paper is divided into sections as follows: In Section II, a brief overview on current PCG techniques has been discussed, followed by an overview on Semantic Web Technologies in Section III. In Section IV and V, some of the overlapping research work from both the fields is discussed along with some related work which is specific to puzzle generation using linked open data. In Section VI, Semantic Based PCG (SBPCG) framework has been proposed and extensively discussed, which combines the strength of both fields and exploit the content of existing knowledge repositories such as DBPedia, WordNet, Freebase, etc. to generate interesting puzzles. Taking the case study of popular word game Hangman, the proposed framework has been analyzed. Finally conclusion and future work has been presented in Section VII.

1 http://dbpedia.org
2 https://www.bbc.co.uk/ontologies
3 http://wordnet.rkbexplorer.com/sparql/

II. PROCEDURAL CONTENT GENERATION IN GAMES : AN OVERVIEW

Procedural Content Generation (PCG) helps in reducing the designers' efforts in creating game content. It can either fully automate content generation or assist designers by generating base artifacts which can be further enhanced by the designer to a full fledged content that can be consumed in a real time environment by player. There are numerous techniques mentioned in [27] and [28], which may be categorized as following:-

Game Creatives : There are the algorithmic approaches to generate various kinds of artifacts or creatives of games such as SpeedTree [38], a commercial tool which uses L-Systems [42] to generate trees and vegetation. Cellular Automata and Binary Space Partitioning method are used for generating mazes and dungeons. Even music content can be generated based on player's current experience [39].

Game Rules and Mechanics : The rules and mechanics are the core parts of the game. These define the actionables for a player to be performed within the game. These include goals, constraints, rewards, punishment, etc. To automatically generate game rules and mechanics, descriptive language such as Video Game Descriptive Language (VGDL) is used, which can easily be manipulated via algorithms to generate new game rules. Ludi System [17] and Angelina project [7] are two popular examples which generate entire game rules and mechanics for Board games and Arcade games respectively.

Game Progression : It is necessary to make the player progress towards mastery. This can be done by dividing the game-play into multiple stages or levels of difficulty. However it is not easy to determine offline precisely that whether the current difficulty level matches the player's capabilities or not. If it is too easy with respect to current player, then player might get bored. If it is too difficult then the player might get frustrated. There are plenty of AI (Artificial Intelligence) techniques which can help in predicting the next game content based on current game state. For example, the famous Min-Max algorithm in *Chess* helps in predicting opponent's next steps by exploring the decision tree, on the basis of which new difficulty level can be decided. Monte-Carlo Tree Search [6] is another algorithm which is more efficient than Min-Max.

Reinforcement Learning is another technique for dynamically adjusting the difficulty by associate every action with some credits (positive or negative). Evolutionary Search Based Strategy [37], uses meta-heuristic approach to generate the new content by performing crossover and mutation over existing content.

Player Modelling and Aesthetics Modeling : In order to make the game adaptive according to players current mood and skills, a predictive model of player is created based on information such as in-Game observations, Player Profile, feedback, etc., so that current player's mood, skills, tactics can be predicted. Pedersen et al. [5] mentioned two approaches for doing this:- (1) Top Down (Model Based) and (2) Bottom Up (Model Free). In Top Down technique, it starts with a hypothetical model with random or most probable weights, then using supervised learning the model get corrected. In bottom up approach, user is put under some category representing user's current aesthetic states using unsupervised learning. These models are then used to generate and evaluate the next set of content.

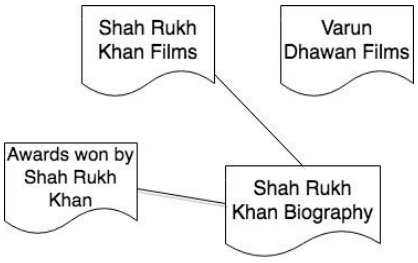
Believable Agents and NPC Behaviour Modelling : Non Player Characters (NPC) plays an important role in many games. They can either act as opponent or assistant. It is necessary for an NPC to believe realistic by human player. For example in a car racing game, it is necessary that the opponent racers (NPC) should race with same skills as human players. Otherwise human player may feel deceived if he smelled that NPC intentionally trying to make human player win; or it may make player frustrated if NPC is exceptionally far better than him. Believable Agents are those which cannot be distinguished by human players that whether the opponent player (or companion) is another human player or a machine. Believable Agents are machines which behaves like a human that is adaptable to player's current state and make player believe that the competent (or companion) is not artificial.

Benchmarking : The generated content or simulator needs to be evaluated, whether it is matching designers and player's expectations or not.

III. SEMANTIC WEB TECHNOLOGIES : AN OVERVIEW

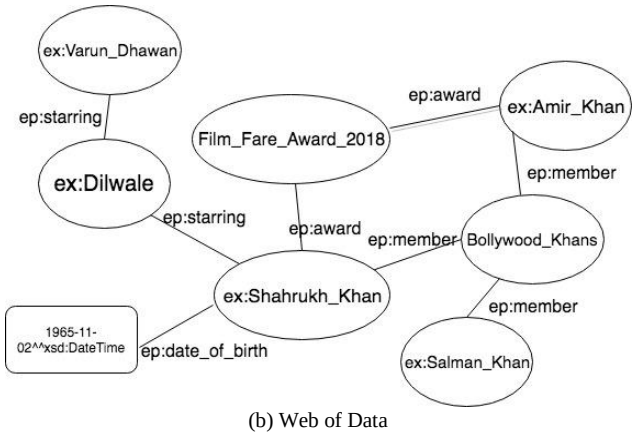
Semantic Web was introduced by Tim Berners Lee in 2001 [1], with the vision that it will replace the current web of documents with web of data and enabling machines to understand the content of web. To understand this better, Fig. 2 shows the two different structures of same information. As can be seen in Fig. 2a, data related to *Bollywood* actor "Shah Rukh Khan" is fragmented in several documents. These documents are linked together, but the granularity of knowledge is restricted to the document level only. Hence, these hyper-linked documents only convey that the documents are associated in some sense. But it doesn't say anything about what part of the document, i.e. which entity of the whole document is actually associated with the entity of other document. On the other hand in Fig. 2b, the data is weaved into a graph, linking individual entities with others along with labeled edges. This structure is having advantage in the sense that a particular user who is searching everything about "Sharh Ruhk Khan", need not have to hover from one document to another. All data may be retrieved easily at one place. Moreover, it also enables to deduce new knowledge which is not explicitly mentioned. For example in fig. 2b, it can be easily deduced that "Shah Rukh Khan" and "Varun Dhawan" are co-actors in film "Dilwale" or how "Shah Rukh Khan", "Amir Khan" and "Salman Khan" are related to each others.

In Semantic Web, the graph is stored in **Resource**



(a) Web of Documents

Description Framework (RDF) which create statements in



(b) Web of Data

Fig. 2: Showing different formats of storing World Wide Web data. Figure 2a showing how current web is storing documents and linking. In Fig. 2b, rather than documents as a whole, the entities within it are linked with other activities which add semantics to the linked data.

the form of triplets: <*subject, predicate, object*>. Hence, for graph in fig. 2b, following triplets can be formed: <*s:ShahRukhKhan p:Film o:Dilwale*> and <*s:VarunDhavan p:Film o:Dilwale*>. The subject and predicate are always in the form of a URI and object can be a URI or a literal of a particular data type as integer, string, datetime, etc. **RDF**

Schema (RDFS) provides basic vocabulary for the basic skeleton under which all entity instances can be stored in some hierarchy. **Web Ontology Language (OWL)** is the extension of RDFS which allows to state additional constraints, such as transitivity relationship, restricted domain of values, cardinality, etc. It is based on description logic and so, brings reasoning power to the semantic web.

SPARQL (Sparql Protoco and RDF Query Language) is a RDF query language which is used to query RDF based data. There are many knowledge bases, which are built on Sematnic Web Technologies such as *DBPedia*[4] – a multi-lingual knowledge base which is developed from the knowledge extracted form Wikipedia, and contains 1.46 billion facts and describes 10 million additional things. Other similar knowledge bases are *WikiData*[5], *Freebase*, various ontologies from *BBC*[6], etc.

Some of the major challenges that Semantic Web Technologies currently facing [4] such as Content Availability, Ontology Availability, Interoperability, Knowledge consistency and evaluation, multi-languae and ambiguity problem, scalability, context aware information retrieval, etc.

IV. SEMANTIC WEB APPLICATIONS IN CONTEXT OF VIDEO GAMES

Semantic Web has a wide variety of applications in general such as supply chain management, media management, data integration, web search and e-commerce, etc. The scope of semantic web in digital games has been identified here as follows:-

A. As storage structure for Games

The RDF can be used to store game data which can be usable in several different games. For examle, Sacco et al. [26] discussed a new ontology to store data related to game characters such as class, role, abilities, skills, power, outfits and relationship with other characters. Similarly, Duric and Konecki [15] created an OWL based ontology for Role Playing Games (RPG) and Tutenal et al. [40] described semantic knowledge framework to store Game World data, such as computational geometry, computer graphics, the rules to design consistent game world.

B. In Computational Creativity

Semantic Web has power to deduce new knowledge. Hence it is possible to generate new artefacts of a particular game from existing game content. Sacco et al. [9] suggested a six step process and various tools to accomplish each step to generate entirely a new game. Those six steps can be grouped into two :1) Information Extraction from various existing sources such as ontologies, web, databases, etc. (2) Merge these resources to create an entirely new game. The author for this use various existing ontologies such as *WordNet*, *DBPedia*, *WikiData*, *GluNet*, *ConceptNet*, along with some game ranking websites such as *GameRanking*, *Games Metacritic*

C. Dynamic Difficulty Adjustment and Game Adaptability

The generated content must be suitable according to player's current skills, abilities and preferences. Lopes et al. [41] proposed a semantic framework to generate adaptive game worlds which generate worlds based on player's current skills and strategies. The author introduces the concept of semantic library which sits on the top of content generator and validate generated content with semantic rules.

D. Player Modelling and Aesthetics Modelling

There is potential in Semantic Web to be used to model player's affective state. Gil, Rosa, et al. [32] and Abaalkhai, Rana, et al. [33] proposed ontological approach to capture human emotions, which can be used to model player's aesthetics.

E. Non Playable Characters (NPC) generation

No Playable Characters are those which act as a companion or competent during the journey in the game. These either help by giving clue, helping hand during combat, or acting as a opponent, or sometimes exist just for aesthetics purpose. Barros et al. [14][10] generated NPC were generated by taking real identities from DBPedia, which reveals some clues during player's journey in Adventure games.

F. Computational Narratives

There are also several attempts have been made to generate linear and non-linear storytelling such as in [29] and [30], which is helpful in generating ontology based stories in games. *GlueNet*[7] is one such English, lexical and common sense database which uses *ConceptNet*[8], *WordNet*[9], *FrameNet*[10] and *VerbNet*[11] for computational storytelling in computer games.

G. Ontologies for Game Studies and Game Generation

Several ontologies have been developed for various purposes. Game Ontology Project (GOP) [23] is the oldest one, which attempt to solve game designers problem who want to study games from several aspects but with some common vocabulary. Similarly Digital Game Ontology [22] combined the Music Ontology, and the Event and Timeline ontology, for describing games. Video Game Ontology [20] aims to support interoperability among several video games and the Game2Web ontology [13] attempted to link in-game events to the entities of real world social data.

4 http://dbpedia.org
5 https://www.wikidata.org
6 bbc.co.uk/ontologies
7 https://graphics.tudelft.nl/glunet/
8 http://conceptnet5.media.mit.edu/
9 http://wordnet.princeton.edu/wordnet/
10 https://framenet.icsi.berkeley.edu/fndrupal/
11 http://verbs.colorado.edu/ mpalmer/projects/verbnet.html

H. *Building Semantic Web using Games*

Not only is Semantic Web helpful in games, but games are also helpful in developing ontologies. *Game with Purpose* [25] is one of the earliest attempts, which builds series of OntoGames in order to weave semantic web, where human labour is exploited with the help of games so that huge amount of data can be linked together with proper consistency and validation, which would otherwise be not so easy. Many other attempts have been made such as in [18] and [16], which uses crowd-sourcing techniques to build knowledge.

a murder mystery game. Another technique is by using traditional graph traversing techniques such as Breadth First Search (BFS), Depth First Search (DFS) and other derivatives, where a node is picked which depicts a broad category in a game such as Movies. Then graph is traversed to it's children and parents up to certain depth to find similar content. Vega-Gorgojo in [11] used this approach using SPARQL endpoint over Simple Knowledge Organization System (SKOS[13]) where concepts explored using *broader*, *narrower* properties of SKOS to DBPedia with depth of 4 or 5, in order to get broad range of topics to generate multiple

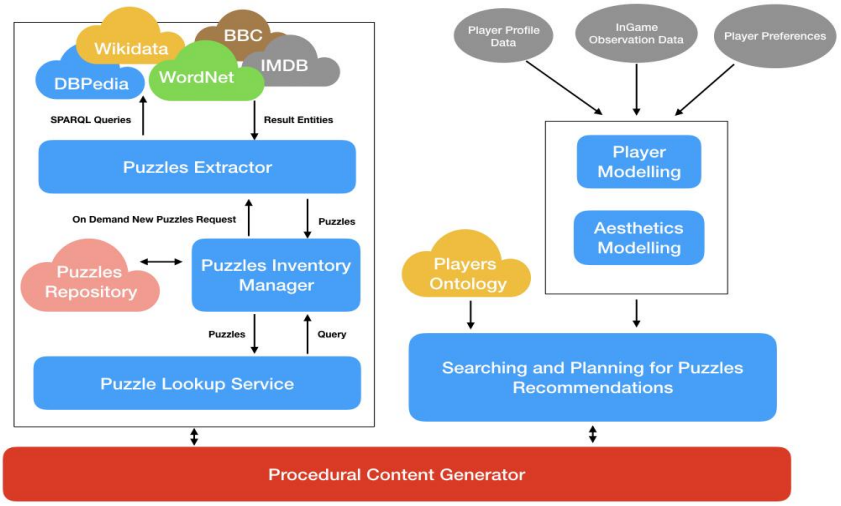


Fig. 3: Proposed SBPCG Framework for Puzzles Extraction showing various major components : Player Modelling, Puzzles Recommendation, Puzzles Extraction and Repository Management.

V. RELATED WORK IN PUZZLES GENERATION FROM OPEN DATA

One of the major challenges in generating puzzles from these knowledge bases is the exploratory nature of the search. That is, how and in what order should the information be explored, so that the latency and computational cost can be minimized while generating good quality of puzzles. One approach is to pick two random interesting nodes from the linked data and crawl through several paths to connect them. This approach is adopted in [14] to generate an adventure game from open data. Here, two entity picked is of type *Person*. Then based on given location coordinates as one of the property of node, these nodes is put on *Open Street Maps*[12]. All the nodes in between them are then put on several coordinates over the map as clues, which is used by the player while traveling the path from source to destination. These clues are generated using description provided at each entity lying in the path connecting earlier picked two entities from open data. Similar approach is taken in [10] to generate choice quizzes. However, this approach doesn't take difficulty into account. Similar multiple choice game puzzles are generated in [21], which evaluate the difficulty by calculating the similarity by evaluating the cosine distance between correct answer and distractors (incorrect answers). The lesser the distance more is the similarity hence more the difficulty for the player to disambiguate. Mmynarz and Zeman in [8] generated Abbreviation guessing quizzes from DBPedia which determine the difficulty of quiz using in-degree and out-degree of the links from particular node.

VI. PROPOSED SEMANTIC-BASED PCG FRAMEWORK FOR QUIZ GAMES

The objective of Semantic Based PCG Framework (SBPCG) is to leverage the strength of Semantic Web Technologies in generating procedural content. The overall procedural content generation process is divided into 3 major components as shown in Fig. 3 : (1) Player Modelling and Aesthetics Modelling, (2) Searching and Planning for

12 https://www.openstreetmap.org/

13 https://www.w3.org/2004/02/skos/

Puzzles Recommendation, and (3) Puzzles Repository Management. The framework is an extension of the work represented by [3] and inspired from the work of Yannakakis and Togelius in [36].

A. Searching and Planning for Puzzles Recommendation

This component takes the current aesthetic state of the player, which determines his current mood and the player model, to determine his current skills and strategies. For example, in Hangman, players gradually develop skills to guess the words by simply guessing vowels first and then most common consonants. Hence it makes the player habitual of using this simple strategy repeatedly which eventually make him bored after some time. Hence in order to break this habit, a proper planning to search the next set of puzzles is required to increase the level of difficulty. For example, in Hangman, for puzzles having very short word length (say 3 or 4) and very low vowel density, player may loose if he guess with same strategy. This component also takes other users' data from all **player's repository** which may be stored in the form of an *ontology*, to identify desired attributes of the puzzles on the basis of *Collaborative Filtering*, which is a well known technique in Recommender systems [31]. Fraihat and Shanibour [34] proposed a framework for Semantic Recommender systems, which can also be used here.

B. Puzzles Repository Management

This component is responsible for extracting puzzles from various knowledge repositories and store these into our own inventory (repository). The puzzle repository can be simple database or it can itself be stored in RDF format. RDF format is recommended as it is scalable and puzzles can be linked to various properties which can help in searching right puzzles for user. Whenever a search query is received from Puzzle Recommender, **Puzzle Lookup service** will search for puzzles with desired attributes in puzzle repository with the help of **Puzzle Inventory Manager**. Puzzle Inventory Manager abstracts the implementation of Puzzle Repository from Puzzle Lookup Service. It is also responsible for initiating *puzzles extraction process* for extracting new puzzles, if the inventory is deficient of any particular category of puzzles demanded. The **Puzzles Extractor** extracts puzzles from knowledge repositories by crawling from node to node with some given strategy and extracts the puzzles from each node visited and sends it to *Puzzle Inventory Manager* to evaluate the puzzles and store it into the repository.

C. Procedural Content Generator

This module provides interface to the client and helps in coordinating with components of framework. Whenever the client requests for next set of puzzles, with player's data as an input, it send the data to player's model to predict player's current skill level and current mood, which is then taken by the *Searching and Planning* process. The planning process involves comparison of current player's data with other player's attributes to extract desired properties for the next set of puzzles. For example, if the current player's profile says that he likes Bollywood movies, then based on matching profiles with other players, it can also be predicted that the player might also like Bollywood songs, and hence we can generate new puzzles. This is known as Collaborative filtering [31]. There is another technique which is known as content-based filtering [24], in which player data is directly used to find similar content which user may like. For example, if a user like Honey Singh's songs, then he may also like Badshaah's songs too, because both singers are *rappers*. When Searching and Planning component determined which category could be best for next set of puzzles, then Puzzle Look Up service is then queried by the generator and it gets the puzzles of desired attributes which are then returned back to client.

1) Overall Content Generation process follows as below:-

1. First, the initial puzzle repository is populated with a wide variety of quizzes in various categories such as Movies, Places, People, etc. using *Puzzle Extraction Process.*
2. When a player start the game, he has to choose certain broad categories of interest from the screen. The preference data along with the player's profile data such as demographics is then sent to the generator.
3. The player's data is then input to the player modelling component, which determines the certain personality traits/skills of the player. If the player has just begin the game, then his current aesthetic state will be default as NEUTRAL, otherwise his current aesthetic state will be predicted.
4. The player's personality traits, skills and preferences are then be used by *Searching and Planning* component, which uses other player's data to identify what kind of puzzles would be most suitable for the current player. This component then output the list of attributes of puzzles that need to be queried in the puzzle repository. For example, if a player's preference is movies and his age is under 12, he falls in Kids category. It will then lookup in **Player's ontology** data to decide what kind of puzzles are best. Hence in this example, all animated movies, kids shows, as a sub-categories of movies would be recommended as one of the attributes.
5. Finally, the attributes of the suitable set of puzzles are queried using the *Puzzle Lookup Service* to fetch the desired puzzles, which are then delivered to player.
6. This process is ongoing during entire gameplay. Whenever the puzzles get exhausted or a player demands new puzzles, new puzzles are fetched using this process.

D. Puzzle Extraction Process

Puzzle extraction from the knowledge repository is a big challenge itself as it is highly dependent on the

availability of content and its quality. Secondly, it is necessary to pre-determine the exploration strategy, otherwise the process will suffer high latency and may produce unfruitful results.

Steps to extract puzzles from knowledge repository:-

1. Initial description of desired attributes for puzzles is gathered such as:
   Category:Bollywood
   Period:1990s
   Labels : movie, actors, awards, etc.
2. These attributes are then given to Puzzle Extractor, which transforms this description into SPARQL queries by looking up appropriate URI of properties for the corresponding attributes.
3. The process starts with the entry point query, which will be given to crawler along with all desired properties.
4. Crawler crawl to those nodes which are having these desire properties. There some other heuristics can also be applied, which evaluate each node to a real valued score for deciding which node to visit first. *Information Foraging* and *Berry Picking* are two important and popular exploratory search techniques [35] that can be used here.
5. At every visit to an entity node, the crawler returns this entity to the puzzle, extractor, which extracts values of the desired properties and using various pre-defined templates for each of the properties, clues are generated. For example if in a movie, a property "Director" is given, then the cue can be "Film directed by XYZ". All permutations of possible puzzles will be generated. Hence if there are N properties found in an entity, then N – 1 properties can be used as clue to guess a particular property.
6. Crawler also marks each node as VISITED, so that it does not traverse the same node again in next finite period of time or unless explicitly directed to.

```
SELECT DISTINCT ?item ?itemLabel ?directorLabel ?castMemberLabel ?producerLabel
   ?pubdate WHERE {
            ?item wdt:P31 wd:Q11424.      %P31 : instance off, Q11424 : film:
            ?item wdt:P577 ?pubdate.      %P577 : publication date,
            ?item wdt:P495 wd:Q668.       %P495: country of origin, Q668:India
            ?item wdt:P364 wd:Q1568.      %P364 : original language of work;
               Q1568 : Hindi
            ?item wdt:P57 ?director.        %pP57 : director
            ?item wdt:P161 ?castMember.  %P161 : cast member
            ?item wdt:P162 ?producer.    %P162 : Producer
            FILTER((?pubdate >= "2017-01-01T00:00:00Z"^^xsd:dateTime) &&
               (?pubdate <= "2017-12-31T00:00:00Z"^^xsd:dateTime))
            SERVICE wikibase:label { bd:serviceParam wikibase:language
               "[AUTO_LANGUAGE],en". }
            }
```

Fig. 4: Sample Query to extract Bollywood Word Puzzles for Hangman

1) Implementation using SPARQL for extracting Puzzles for Hangman

For puzzle's extraction in case of Hangman, the final puzzles should be in the form of a *word, hint, tag*. Consider a scenario that a player likes to play movie quizzes. Based on the player's personality traits/skills and preferences, the *Searching and Planning component,* may recommend that movie quizzes with attributes such as *actors, film producer, director, film title,* would be suitable for this player. Hence, the SPARQL query shown in Fig. 4 can be used to extract and generate puzzles. When this query is executed, Puzzle Extractor extracts around 500 puzzles with all the possible permutations as shown in Table 1. From the results, it can be observed that all the possible puzzles are generated. For example, to guess "Rohit Shetty" multiple clues are generated. Similarly to guess "Kung Fu Yoga", all properties are combined together to form a single clue. This also helps in determining the level of difficulty of puzzles based on number of clues available to guess a word.

Table 1: Result Output of Puzzle Extractor (showing a portion)

Word	Hint	Tag
Golmaal 4	Directed By Rohit Shetty, Starring Ajay Devgan, Produced by Dhillin Mehta	Movie
Ajay Devgan	Star Cast in Golmaal 4	Actor
Rohit Shetty	Directed Golmaal 4	Film Director
Kung Fu Yoga	Directed By Stanley Tong, Starring Sonu Sood, Produced by Jackie Chan	Movie
Sonu Sood	Star cast in Kung Fu Yoga	Actor
Stanley Tong	Directed Kung Fu Yoga	Film Director
Stanley Tong	Producer of Kung Fu Yoga	Film Producer
Sonu Sood	Star cast in Kung Fu Yoga	Actor
Stanley Tong	Directed Kung Fu Yoga	Film Director

E. Managing Diversity

It is important to have diversity in the puzzles, otherwise playing the same kind of puzzles may makes a player bored soon.

But diversity should not only be measured at the time of puzzles extraction, but also at the time of final content delivery to the player during gameplay. Because, unlike recommendation of products in burst at some e-commerce website, in games, it is the ordering of puzzles which the players observe. For example, if a player is guessing movies in every puzzle for the same kind of clues such as "starring :

Amir Khan", "Starring : Varun Dhawan", etc then player will not find the puzzles diverse. But if the puzzles are asked with variety of clues such as sometimes player has to guess movie, sometimes movie actor, sometimes the shooting place, etc., then the puzzles will be observed as diverse in variety. Hence ordering is what matters here. To manage diversity, distance among these entities can be computed. One approach is by computing cosine angle, where the feature vector of properties of puzzles is created and dot product is evaluated which returns a value in between 0 and 1. 0 means the puzzles are the most similar and 1 means they are completely different (orthogonal).

Another way is to calculate the semantic distance between two nodes in RDF itself by counting number of nodes in path in between two desired entities. But it may note be feasible due to higher latency and computation to calculate similarity in real time. So the best way seems here to annotate each puzzle with some meta data at the time of puzzles extraction, which depicts puzzles properties. Then, at real time, the cosine distance between a fixed set of puzzles in memory is calculated. The puzzle having this similarity distance S greater than some threshold T will be eligible to be next in the queue of puzzles.

In order to get broad variety of Bollywood Puzzles during extraction, some generic starting point need to be given to the crawler rather than specific, such as with following SPARQL query:-

```
SELECT ?S ?P {
   ?S ?P dbc:Bollywood .
} LIMIT(1000)
```

which results in distinct entities related to Bollywood category as shown in Fig. 5 (showing only few entities). A good diversity can be observed in this result, such as Actors/Actresses, Film Studios, Awards, Theater shows, Film Production houses, famous personalities, whichever falls under Bollywood category, all at one place. Now it's the crawling strategy how these entity nodes to be explored and generate puzzles at the visit of each of the nodes.

F. Managing Non Repetitive Content

Repetitive content may or may not be desirable. For example, in casual games, it is required to not repeat the content or at least should not be completely identical. In other cases such as in serious games, the intention is to train the people about a particular subject , where repetition may be required. To manage non-repetitive content, an additional time-stamp can be added to each puzzle corresponding to a particular a player, so that it can be prevented for being played again if total duration since last played is less than some threshold time.

```
s
http://dbpedia.org/resource/Pakshiraja_Studios
http://dbpedia.org/resource/Coronation_Cinema
http://dbpedia.org/resource/List_of_Bollywood_actresses
http://dbpedia.org/resource/Samir_Bhamra
http://dbpedia.org/resource/Dharma_Productions
http://dbpedia.org/resource/RDB_(band)
http://dbpedia.org/resource/Walk_of_the_Stars
http://dbpedia.org/resource/Zangoora
http://dbpedia.org/resource/Aamir_Khan_Productions
http://dbpedia.org/resource/Mirchi_Music_Awards
http://dbpedia.org/resource/Jaideep_Chopra
http://dbpedia.org/resource/Teji_Bachchan
```

Fig. 5: Result output, showing a portion of Bollywood entities for SPARQL query executed over DBPedia

G. Puzzle Extractor Plug-ins

Sometimes, a single Puzzle Extractor might not be enough to generate a variety of content, as it may be easy to extract puzzles in the same way for every kind of puzzle, because the data may be fragmented in several different repositories, documents, databases, XML, etc. Hence extraction process may be different. Extractor Plugins provides additional capabilities to the puzzle extractor which deals with specific problems of puzzle extraction from various sources, generate interesting clues based on it's own specialization. For example, a Bollywood Puzzle Extractor plugin may generate more interesting clues for Indians as it is configured better to Indian languages, humor, culture, etc., which a general extractor may not be capable of. These plug-ins can be integrated with Puzzle Extractor Tool to extend it's functionality. Hence whenever a particular category of puzzles are demanded, Puzzle Extractor first lookup for right plug-in installed and assign the responsibility of extraction process to that plug-in.

H. Puzzles Explorer : Manual Puzzles Extraction

Sometimes, it is not desirable to extract every possible puzzle, as this can cause a lot of undesirable puzzles which may not seem interesting. Also there may be some other interesting properties found in the journey, which may have been overlooked by the crawler or, the clues generated by puzzle extractor can be improved with human assistance. In these cases, human intervention is necessary. Hence, before adding extracted puzzles into the repository, a designer can evaluate the puzzles in some User Interface tool called Puzzles Explorer, where designers can evaluate, filter out, edit puzzles before adding them into repository. A designer can also control the crawler direction to direct which node to be visited next. If any new interesting property is found in the visited node, user can immediately add that property with corresponding templates in crawling navigation strategy. This enables the human assisted puzzle generation process.

VII. CONCLUSION AND FUTURE SCOPE

In this paper, the two most emerging research fields has been explored. One is the Semantic Web, which attempts to organize the data into structured format for better accessibility to both humans and machines. Another is Procedural Content Generation which is attempting to make machines computationally creative. From the research work

presented in this paper, following advantages of Semantic Web in video games can be concluded:-

1. By leveraging the power of semantic web, accessibility of game content may be improvised. Hence by storing game content in RDF, it become easy to retrieve similar game content while still maintaining the diversity. This is one of the most desiring property of any game content generator.
2. Semantic Web has power of deducing new knowledge from existing knowledge. Hence it may significantly support existing PCGs in enhancing computational creativity.
3. Scalability, interoperability and machine understandability are key strengths of Semantic Web, which when applied in digital games may serve several distinct games from single source of knowledge repository and can be scaled to many folds seamlessly.

Hence, by looking into these advantages, a Semantic Based PCG framework has been propsed which connects parts of existing PCG (Player modelling, Searching and Planning) with Semantic Web Technologies such as Player's Ontology, RDF based puzzle repository, open knowledge repositories (DBPedia, WordNet, Wiki data, etc). Various problems and their solutions have also been discussed that may challenge the proposed framework. This framework may covers a wide range of quiz based games such as Hangman, Crossword, Word-Search, Movie Quiz, Logo Quiz, etc. It not only consider the puzzles extraction and repository management part, but also considers how to deliver the game content that keeps the player engaged.

This framework can be extended or modified to cover other game genres also. It can be used to generate not only quiz puzzles, but also the new game rules, mechanics, characters, sounds, etc. These can be extracted and added to the puzzle repository too, or a new repository can be created for it. So, whenever a quiz needs to be delivered to the player, same rules, goals and mechanics which are balanced with the quiz, can also be delivered during the game-play. This enhances the dynamic experience of the player and the diversity of the puzzles. Puzzle Extraction process greatly determine the quality of puzzles. Though it is easy to build a crawler and generate puzzles using pre-defined templates, it is still far from the quality of puzzles generated by humans. The clue creation by machines still does not match the creativity of humans. This leaves open scope for future research work in this direction. Using Puzzles Explorer, the puzzles quality can be improved. Though it is not completely automatic, but will greatly reduce manual puzzle generation. Semantic Web may also help in identifying valid starting point for character generation, game worlds, etc. It may support in making good heuristics while exploring decision trees or by tuning fitness functions in evolutionary algorithms while searching for game content. Hence studying PCG and Semantic Web Technologies together crries great scope for future work in the field of game development which may provide more robust and generic solutions as compare to existing solutions.

REFERENCES

[1] T. Berners-Lee, J. Hendler, and O. Lassila, “The semantic web”, Scientific American, Vol. 284, Issue 5, pp. 28-37, 2001.

[2] N. Shadbolt, T. Berners-Lee & W. Hall, “The semantic web revisited”. IEEE intelligent systems, Vol 21(3), pp 96-101, 2006

[3] V. Vashistha, “Solving Content Generation Problem in Hangman using Search Based Strategy”, In Proceedings of the 12th INDIACom; INDIACom-2018; IEEE 5th International Conference on “Computing for Sustainable Global Development", India, 2018

[4] R. Benjamins, J. Contreras, O. Corcho, & A. Gomez-Perez. “The six challenges of the Semantic Web”, 2002

[5] C. Pedersen, J. Togelius, & G. N. Yannakakis, “Modeling player experience for content creation”. IEEE Transactions on Computational Intelligence and AI in Games, Vol 2(1), pp 54-67, 2010

[6] C. B. Browne, E. Powley, D. Whitehouse, S. M. Lucas, P. I. Cowling, P. Rohlfshagen, ... & S. Colton. “A survey of monte carlo tree search methods”. IEEE Transactions on Computational Intelligence and AI in games, Vol 4(1), pp 1-43. 2012.

[7] M. Cook, & S. Colton, “Multi-faceted evolution of simple arcade games”. In Computational Intelligence and Games (CIG), 2011 IEEE Conference on, pp. 289-296, 2011

[8] J. Mmynarz, & V. Zeman. “DB-quiz: a DBpedia-backed knowledge game”. In Proceedings of the 12th International Conference on Semantic Systems, ACM, pp. 121-124, 2016

[9] O. Sacco, A. Liapis, & G. N. Yannakakis, “A holistic approach for semantic-based game generation.” In Computational Intelligence and Games (CIG), IEEE Conference on, IEEE pp. 1-8, 2016

[10] G. A. Barros, A. Liapis, & J. Togelius, “Murder mystery generation from open data”. In Proceedings of the Seventh International Conference on Computational Creativity, 2016

[11] G. Vega-Gorgojo. Clover Quiz: a trivia game powered by Dbpedia.

[12] H. Liang, S. Deng, J. Chang, J. J. Zhang, C. Chen & R. Tong, “Semantic framework for interactive animation generation and its application in virtual shadow play performance.” Virtual Reality, pp 1-17, 2018

[13] O. Sacco, M. Dabrowski, & J. G. Breslin, “Linking in-game events and entities to social data on the web”. In Games Innovation Conference (IGIC), 2012 IEEE International, IEEE, pp. 1-4. IEEE, 2012

[14] G. A. Barros, A. Liapis & J. Togelius, “Playing with Data: Procedural Generation of Adventures from Open Data”. In DiGRA/FDG, 2016

[15] B. O. Duric & M. Konecki, “Specific owl-based rpg ontology”. In Central European Conference on Information and Intelligent Systems, pp 185, 2015

[16] C. Sarasua, E. Simperl & N. F. Noy, “Crowdmap: Crowdsourcing ontology alignment with microtasks”. In International Semantic Web Conference, Springer, Berlin, Heidelberg., pp. 525-541, 2012

[17] C. B. Browne ”Automatic generation and evaluation of recombination games”, Doctoral dissertation, Queensland University of Technology, 2008.

[18] T. Markotschi & J. V□blker “GuessWhat?!{Human intelligence for mining linked data”, 2010.

[19] J. Togelius, A. J. Champandard, P. L. Lanzi, M. Mateas, A. Paiva, M. Preuss & K. O. Stanley. “Procedural content generation: Goals, challenges and actionable steps”. In Dagstuhl Follow-Ups. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, Vol 6, 2013

[20] J. Parkkila, F. Radulovic, D. Garijo, M. Poveda-Villalón, J. Ikonen, J. Porras, J & A. Gómez-Pérez, "An ontology for videogame interoperability". Multimedia Tools and Applications, Vol 76(4), pp 4981-5000, 2017
[21] C. Lin, D. Liu, W. Pang, & Z. Wang. "Sherlock: a semi-automatic framework for quiz generation using a hybrid semantic similarity measure". Cognitive computation, Vol 7(6), pp 667-679, 2015
[22] J. T. Chan & W. Y. Yuen. "Digital game ontology: Semantic web approach on enhancing game studies". In Computer-Aided Industrial Design and Conceptual Design, 2008. CAID/CD 2008. 9th International Conference on (pp. 425-429). IEEE, 2008
[23] J. P. Zagal & A. Bruckman, "The game ontology project: Supporting learning while contributing authentically to game studies". In Proceedings of the 8th international conference on International conference for the learning sciences. International Society of the Learning Sciences. Vol. 2 , pp. 499-506, , 2008
[24] R. Van Meteren & M. Van Someren, "Using content-based filtering for recommendation". In Proceedings of the Machine Learning in the New Information Age: MLnet/ECML2000 Workshop, pp. 47-56, 2000
[25] K. Siorpaes & M. Hepp. "Games with a Purpose for the Semantic Web". IEEE Intelligent Systems, Vol 23(3), 2008
[26] O. Sacco, A. Liapis, & G. N. Yannakakis, "Game Character Ontology (GCO) A Vocabulary for Extracting and Describing Game Character Information from Web Content". In Proceedings of the 13th International Conference on Semantic Systems, ACM. pp. 9-16., 2017
[27] N. Shaker, J. Togelius & M. J. Nelson. Procedural content generation in games. Springer International Publishing, 2016
[28] G. N. Yannakakis, J. Togelius. Artificial Intelligence and Games. Springer, 2017
[29] D. Winer, "Review of Ontology Based Storytelling Devices". In: Dershowitz N., Nissan E. (eds) Language, Culture, Computation. Computing of the Humanities, Law, and Narratives. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, vol 8002. 2014
[30] S. Göbel, F. Mehm, S. Radke, & R. Steinmetz, "80days: Adaptive digital storytelling for digital educational games". In Proceedings of the 2nd international workshop on Story-Telling and Educational Games (STEG'09), Vol. 498, No. 498, 2009
[31] X. Su, & T. M. Khoshgoftaar "A survey of collaborative filtering techniques". Advances in artificial intelligence, 2009.
[32] R. Gil, J. Virgili-Gomá, R. García, & C. Mason, "Emotions ontology for collaborative modelling and learning of emotional responses". Computers in Human Behavior, Vol. 51, pp 610-617, 2015
[33] R. Abaalkhail, B. Guthier, R. Alharthi & A. El Saddik, "Survey on ontologies for affective states and their influences". Semantic web, (Preprint), pp 1-18, 2017
[34] S. Fraihat & Q. Shambour. "A framework of semantic recommender system for e-learning". Journal of Software, Vol. 10(3), pp 317-330, 2015
[35] R. W. White & R. A. Roth, "Exploratory search: Beyond the query-response paradigm". Synthesis lectures on information concepts, retrieval, and services, Vol. 1(1), pp 1-98, 2009
[36] G. N. Yannakakis & J. Togelius "A panorama of artificial and computational intelligence in games". IEEE Transactions on Computational Intelligence and AI in Games, Vol. 7(4), pp 317-335, 2015
[37] J. Togelius, G. N. Yannakakis, K. O. Stanley & C. Browne. "Search-based procedural content generation: A taxonomy and survey". IEEE Transactions on Computational Intelligence and AI in Games, Vol. 3(3), pp 172-186, 2011
[38] Visualization, I. D. (2014). Speedtree.
[39] D. Plans, & D. Morelli, "Experience-driven procedural music generation for games". IEEE Transactions on Computational Intelligence and AI in Games, Vol. 4(3), pp 192-198., 2012
[40] T. Tutenel, R. M. Smelik, R. Bidarra, & K. J. de Kraker. "Using Semantics to Improve the Design of Game Worlds". In AIIDE, 2009
[41] R. Lopes & R. Bidarra. "A semantic generation framework for enabling adaptive game worlds". In Proceedings of the 8th International Conference on Advances in Computer Entertainment Technology,. ACM, pp 6, 2011
[42] P. Prusinkiewicz, "Graphical applications of L-systems". In Proceedings of graphics interface, Vol. 86, No. 86, pp. 247-253, 1986

Authors' Profile

Mr. Vivek Kumar Vashistha completed his Masters in Technology (Information Technology) in 2018 from University School of Information Communication and Technology, GGSIP University, Delhi, India. He has 5 years of professional experience in IT industries and developed many educational Games and other genres for various popular mobile platforms. His main research interest focuses on Procedural Content Generation in Games, Entertainment Computing and Computational Creativity. He has published a research paper in IEEE international conferences in same area..

Dr. Sanjay Kumar Malik completed his Ph.D. in the area of "Semantic Web" from USIC&T, GGSIP University, Delhi. He is currently working as Associate Professor in University School of Information, Communication and Technology, GGSIP University. He has more than 18 years of industry and academic experience in India and abroad (Dubai and USA). His areas of research interest are Semantic Web and various Web Technologies. He has several research papers in reputed international Conferences/ Journals (IEEE/ Springer). He has been session chair in several international conferences and technical committee member of several conferences. He has been honoured with third best researcher award, 2011 by GGSIP University for his contributions in research.

