Learning Preposition Priors to Generate Scene from Text Using Contact Constraints

S. Yashaswini and S. S. Shylaja

Abstract — In this paper, we propose the method of generating a 3D scene from text with respect to interior designing by considering the orientation of every object present in the scene. Thousands of interiors designing related sentences are generated using RNN to preserve context between sentences. The BiLSTM-RNN-WE method is used for POS Tagging, blender is used to generate 3D scene based on query. This paper focuses on interior designing and has considered objects placement with respect to the preposition in the Sentence. Our approach uses Natural Language processing to extract useful information from the user text, which will aid the rendering engine generate better scene.

Key words — BiLSTM, Natural Language Processing, RNN.

I. INTRODUCTION

People communicate their feelings, thoughts, and ideas through language, so this is an attempt to mimic imagination of their dream home through visualization. One such goal is to bridge the gap between natural language and visual modality.

The task of generation of a scene can be interpreted mainly in two ways. First method is the simple drag and drop of individual models to suit the requirement of the user. This will meet the entire orientation aspect of the user. But for the end user, the task of scene design can be very complex using the drag and drop method as there are many models to search for and it becomes a cumbersome task for the correct scene.

The second approach would be the skill to visualize and describe the scene using English sentences which are then mapped to our models.

The ability to generate a scene with rudimentary English sentences can be simpler as the user just has to describe his visualization and the task of placing and selecting the Objects are done by our model. We consider the second approach, and the task of the end user is minimized to a great extent.

Text to 3D scene generation has a wide number of applications. It can be used in the educational sector where a complex concept could be described, and a 3D geometric model could explain the concept.

3D scene generation can be useful even in fields like art, forensics, computational design, fabrication, and augmented reality. In this paper, the main focus is on interior designing which has a lot of focus on the orientation of the objects.

Imagine describing arrangements in the living room as “there is a table next to the chair”. After preprocessing the sentence is POS tagged using Bi directional long short term memory-Recurent Neural Network-Word Embedding (BLSTM-RNN-WE) approach.

The noun and preposition are extracted matched with model ID from ShapeNet dataset and displayed in a scene using blender as shown in Fig. 1. We currently have 3d shape databases like Turbosquid, 3Dwarehouse, Yobi3D, and 3D modeling software’s like Maya, 3DS MAX which is expensive and time consuming for manual designs, hence this is an approach to automatically generate scene from natural language text descriptions.

The key components of the approach considered is capturing all the prepositions and mapping the preposition with the respective parent-child relationship.

Considering the orientation aspects of every object based on the requirements given by the end user, a 3D model of the scene is generated and presented. An implied parent is considered if it is not explicitly stated by the user.

Fig 1. Scene generation from natural language text descriptions.
Most recent work on Text to 3D scene Generation has been carried out by Stanford University [1]-[4] which considers objects and trains a classifier on a scene discrimination task and extracts high-weight features that ground lexical terms to 3D models. They have integrated their learned lexical groundings with a rule-based scene generation approach and have shown through a human judgment evaluation that the combination outperforms both approaches in isolation.

The other work in this field is done by WordsEye specified by Bob Coyne, Richard Sprout [5] which used manually chosen mappings between language and objects in scenes. WordsEye has used natural language as a medium for describing visual ideas and images to acquire artistic skills on window-based interface, it automatically converts text into 3D-scenes by depicting entities and objects involved, their poses, grips, shapes and spatial tags and relations, color, kinematics, attributes like twisting, bending and tries to avoid conflicting constraints by specifying path, orientation, and position as specified by Bob Coyne, Richard Sprout [5]. But these approaches when scaled down to interior designing fails to give good results for objects whose reference is a wall or the ceiling, neither does it consider the orientation aspect of objects.

II. IMPLEMENTATION

The task of generating a 3D scene involved cleaning of the data that the text processor processes. This cleaned data is given to the text parser which gives all the parent-child relationship of the 3D models required [6]. The output generated is later used to render a 3D scene.

A. Preprocessing:

There were several types of noise that were encountered during text processing, and each had to be resolved differently for different purposes. There were some modifications that were significant.

1) Corrections in the Punctuation

For punctuations like full stop and comma, the text parser considered the word with the punctuation and took it as a noun, e.g., “There is a chair, a table and a sofa.” Here, “chair,” was processed as a noun instead of considering it as “chair” “,” which is the required format.

2) Numerical Representations in the result

For numbers written as words, e.g., “three”, it had to be converted into numerical “3”. This is useful for the parser to know how many objects were considered in case of plural forms of the word [7].

B. Parts of Speech Tagging

Bidirectional Long Short-Term Memory Recurrent Neural Network (BLSTM RNN) has been shown to be very effective for tagging sequential data, e.g., speech utterances or handwritten documents [8]. The usage is illustrated in Fig. 2.

In this approach, BLSTM RNN is also used to do a tagging task, but only has two types of tags to predict: incorrect/correct. The input is a sequence of words which is a normal sentence with some words replaced by randomly chosen words. For those replaced words, their tags are 0 (incorrect) and for those that are not replaced, their tags are 1 (correct). Although it is possible that some replaced words are also reasonable in the sentence, they are still considered “incorrect”. This requires certain modification of tags.

![](image-url)

Fig. 2. BLSTM-RNN for POS Tagging.

1) Modifications of tags

When the main objective is the orientation of objects, some of the prepositions like ‘top’ and ‘front’ were tagged as noun (NN), which was not suited for interior designing.

Hence, the next task was to change some of the tags, e.g., “A lamp on top of a table.” Here, top is a preposition rather than the noun form. These tags had to be changed to their appropriate POS tag (Preposition). Some prepositions like “in front of” and “next to” were taken as separate words instead of a single preposition. This was revised in the text processor.

C. Task Description

The preprocessed text was now fed to the text processor and the desired output was generated.

1) Model characteristics

Since all the models were singular in nature, the presence of a plural word in the given input would not map the word to the corresponding model. To avoid this complication, stemming and lemmatizing of the words were done. For example, “Chairs”, which is the plural form, is considered as “chair” in the final output of the processor [9].

2) Text processor features

One of the outputs of the processor contained the word followed by the stemmed, lemmatized form of the word (which was used for getting all the nouns), the POS tag, the Named Entity Recognition (If present), and the parsed output.

The output of the text processor also contained tokens, expression parse trees, parent-child relation (if present). From this, the final output of the text processor considered all the nouns that were present in the sentence and the parent-child relationship.

3) Mapping of model with the output of the text processor

From the text file, a suitable object had to be chosen from the list of models obtained from shapenet. The models were organized based on Wordnet’s Synset (Synonym Set) ID [11].

The output of the processor contained a Synset (Synonym Set) ID of the models. A suitable object was chosen from this synset ID. All the 3D models containing the particular noun in the input were considered. Every model had tags associated with them. Hence the tag should be mapped with appropriate model as shown in Fig. 3.

There were many models that could be considered to the particular noun. Some models had many tags. This reduces the probability of it being the model that the user expects.
Given a sentence $w_1, w_2, ..., w_n$ with tags $y_1, y_2, ..., y_n$, BLSTM RNN is used to predict the tag probability distribution of each word.

The input vector $I_i$ of the neural network is computed as:

$$I_i = W_1 w_i + W_2 f(W_i)$$

Where $W_1$ and $W_2$ are weight matrixes connecting two layers. The word embeddings are initialized with uniformly distributed random values, ranging from -0.1 to 0.1. The implementation of BLSTM layer is skipped in this paper. It outputs the tag probability distribution of input word $W_i$. All weights are trained using back propagation and gradient descent algorithm to maximize the likelihood on training data:

$$Y_i \in 1, ..., n$$

$$P(y_i | w_1, w_2, ..., w_n)$$

The Units in the metrics are probability varying from 0 to 1. The value 0 indicating the least probability and value 1 indicates the highest probability.

III. RESULTS AND DISCUSSIONS

The noun extracted from the sentences helps to identify the object but placing the objects requires the spatial knowledge and support constraints.

The Tkinter is used where the dropdown menu is created to choose the objects with respect to the placement constraints like left_of, right_of, next_to, front, above, below. The sentence selected through the interface are tokenized, POS Tagged and the preposition is extracted to learn semantic inference as shown in Fig. 6.

The semantic learned is then displayed with the objects vase and table, from Fig. 7 it is clear that the scene generated is not accordance with the human intuition. The vase is not all placed on the table but its flying in air without the contact constraint.
Therefore, it is very essential to learn the priors for scene generation like placement, contact constraints and realizing the semantics of preposition w.r.t to other words in sentence. The scene generated for the sentence “The vase is on the table” as selected from the GUI interface as shown in Fig. 8.

The Scene generated by choosing appropriate preposition implied that the contact constraint is met, the scene clearly shows that the vase and table surfaces are touched as shown in Fig. 9.

![Fig. 8. Sentence generated by changing preposition.](image)

![Fig. 9. Scene generated for “The vase is on the table”.](image)

IV. CONCLUSION

The experimentation carried out shows that the rogue scores outperform by considering the n overlap n-grams. The meteor scores are zero irrespective of n iterations due to the dataset lacking fluency and adequacy. The interior design dataset is generated by applying the RNN-LSTM model to human-annotated sentences. The dataset is generated and not translated hence it is hard to obtain fluency and adequacy.

V. DRAWBACKS

There were many erroneous results from the final text processor output. For sentences such as “There is a chair in front of a table. There is a sofa in front of the chair.” There should have been two instances of “in_front_of” with different nouns, but there was only one instance of “in_front_of” which mapped to one of the sentences. There is a similar case for the preposition “next_to”. There were limitations with respect to the quantity of a particular object. The limit for the quantity of objects is 20. If the user wants an object more than twenty times, it is not considered. The text processor works for few limited sentences at a time [18]. For objects that had to be placed on a table, if there was lack of space, it would not be placed in an appropriate position. Bounds checking for individual object was not done in these cases.

![Fig. 10. Scene generated for “The Vase is in front of table”.](image)

The experimentation was carried out for other sentences like “the vase is in front of the table”. The result got was unexpected the vase and table were of same size and more over the object vase occluded the table completely as shown in Fig. 10. It is learned that the size and distance are also important concerns while generating the scene prior and hence the attempt is made to separate the object from a distance as shown in Fig. 11.

![Fig. 11. Scene generated for “The Vase is in front of table” after entering the distance of separation.](image)
VI. FUTURE ENHANCEMENTS

When the user enters more than three sentences, the result can be incorrect. The enhancement plan is to make it work for a complex set of sentences. At present, the same object is always selected. If the object selection is randomized, the result may be incorrect. Hence, a list of appropriate objects should be made. Better object collision and bounds checking with respect to every object needs to be implemented [19]. Rule based approach can be used to identify the prepositions.

Neural network can be used to train the objects and placing them to scene more models can be generated using GAN [20]. A good metric could be used to measure accuracy of generated scene.

REFERENCES

[1] Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization, pp. 65–72.

[2] George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In Proceedings of the second international conference on Human Language Technology Research, pp. 138–145.

[3] Morgan Kaufmann Publishers Inc. Hideki Iossaki, Tsutomu Hirao, Kevin Duh, Katsuhibo Sudo, and Hajime Tsukada. 2010. Automatic evaluation of translation quality for distant language pairs. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pp. 944–952. Association for Computational Linguistics.

[4] Guillaume Klein, Yoon Kim, Yunian Deng, Vincent Nguyen, Jean-Sebenllart, and Alexander M. Rush. 2018. Opennmt: Neural machine translation toolkit.

[5] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. Text Summarization Branches Out.

[6] Nitin Madnani. 2011. iblue: Interactively debugging and scoring statistical machine translation systems. In 2011 IEEE Fifth International Conference on Semantic Computing, pp. 213–214. IEEE.

[7] Kishore Papineni, Salim Roukos, Todd Ward, and WeiJing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on association for computational linguistics, pp. 311–318. Association for Computational Linguistics.

[8] Maja Popovic. 2015. chrf: character n-gram f-score for automatic mt evaluation. In Proceedings of the Tenth Workshop on Statistical Machine Translation, pp. 392–395.

[9] Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Mucciulla, and John Makhouli. 2006. A study of translation edit rate with targeted human annotation. In Proceedings of the association for machine translation in the Americas, volume 200.

[10] Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144.

[11] Qi Ye, Sachan Devendra, Felix Matthieu, Padmanaban Sarguna, and Neuhig Graham. 2018. When and why are pre-trained word embeddings useful for neural machine translation. In HLT-NAACL.

[12] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. arXiv preprint arXiv: 1904.09675.

Yashaswini S. is presently Working as Assistant Professor in Department of CSE, Cambridge Institute of Technology, Bangalore, she is having 8 years of Teaching Experience and almost 2 years of industrial experience. She is Currently pursing her PhD in the text to scene generation under VTU Belagavi.

She has published many research papers which includes 6 international journal, International conference and 4 national conference papers, she has proactively participated in many FDP’S and workshop, as a part of industrial interaction has participated FDP on “Business Process Modeling Tools and Process Model” at J.P. Morgan PVT LTD and “Cyber Forensic workshop at NCS” at Nihon Communications PVT LTD. Her Areas of Interests include Digital Image Processing, NLP, Machine Learning, Computer Networks, cloud computing, and Information Security.

Dr. Shylaja S. S. is the Chairperson, Department of Computer Science and Engineering (UG Studies) at P E S University, Bangalore. She is also the Professor and Head, Department of CSE and Department of ISE at P E S Institute of Technology (PESIT), Bangalore. She completed her Bachelor’s degree from UVCE, Bangalore University in 1989, Master’s degree from Sri Jayachamarajendra College of Engineering, Mysore University in 1993.

She secured the First rank in the university in her M. Tech course. She completed the ‘C’ level course from DOEACC and holds a Ph.D. in the domain of Face Recognition from Visvesvaraya Technological University (VTU). She has 28 years of teaching experience and 15 years of research experience.

She has several journal articles, national and international conference publications to her credit. She has taught a plethora of courses and has guided several projects and research activities at undergraduate and postgraduate levels. She is a member of BOS of several organizations, reviewer of conferences, and member of many technical committees.

Dr. Shylaja is heading two centers at P E S University, the Center for Data Sciences and Applied Machine Learning (CDSAML) and the Huawei Innovation Lab, facilitating research and internship opportunities for students and faculty members. Her research areas include Image Processing, Computer Vision, Natural Language Processing, and Machine Learning.