Generation of floor plan using its Textual Description

Mahak Jain (B14CS042) Anurag Sanyal (B14CS007)
Mentor : Dr. Chiranjoy Chattopadhyay
Department of Computer Science and Engineering
Indian Institute of Technology, Jodhpur

Motivation: Eliminate the necessity of CAD and generate an image, based entirely on description in layman terms.

Objective: We aim to generate the image of the floor plan given its textual description. The input can be multiple descriptions of a floor plan whose image is to be generated.

Challenges:

- Insufficient data to train learning models.
- The textual descriptions are too varied and it is difficult to come up with a generalized format of description for training the model.

Methodology:

1. **Data Collection & Preprocessing –** In this step, we surveyed 4 different descriptions each for 40 images. We collected 4 descriptions of 4 different images from a single person, thus a total of 160 descriptions from 40 people. Then, arranged the descriptions of a single image together.

   Then, we preprocessed the data to feed our training model. Preprocessing includes -

   - converting into lower case
   - pronoun replacement
   - removing extra whitespaces
   - removing stopwords

   We also prepared the **output summary** for each image manually to train a learning model for text summarization but it could not be used because the data was too less to employ learning methods for summarization.

2. **Text Summarization –** There are two approaches to summarize text – extractive and abstractive. Extractive summarization uses statistical approach for selecting important sentences or keywords from document and concatenate them to form a summary. The abstractive approach generates a summary that keeps original intent. It is more like how humans summarize.

   We have used a technique of **extractive** Text Summarization in which description is preprocessed, undirected graph is constructed to calculate similarity between sentences, a

word class is attached to each word, sentences are ranked according to word class similarity among sentences and top ranked sentences are included in the summary. This method is very similar to Page Ranking Algorithm. We take reduction ratio as input in this step.

Advantages -

- Easier to work with and does not require learning.
- Easy to compute because it does not deal with the semantics.

Limitations -

- Lack flexibility and cannot paraphrase. Example - If there is no sentence that explicitly describes the location of a particular room, we will not be able to extract its location.
- Results in lengthy summary.
- Suffers from inconsistencies when presented with conflicting data.

Despite its limitations, the reasons we cannot implement an abstractive approach are -

- It has extremely high hardware demands.
- We have insufficient data to train such a model.
- It is more complex to implement than extractive techniques.

3. **Sentence Classification & Clustering** – The summary generated in previous step needs to be rearranged. Thus, we convert words into vectors using **word2vec** and a **Binary CNN** (Convolutional Neural Network) classifier is trained for each room to label each sentence with a room which most likely it describes. Then, we cluster sentences room-wise.

   The data for each room used to train the classifier is **scraped from internet** as well as from the descriptions collected through survey. The rooms for which we have trained Binary CNN are -
   - bedroom
   - kitchen
   - bathroom
   - hall
   - dining area
   - relation

   There is an additional CNN model trained to label **relational sentences.** The sentences which describes adjacency or door connectivity relationship between multiple rooms are defined as relation sentences. Example - “The bedroom leds to bathroom.”

   The reason we used binary classifier is that there can be sentences which describe multiple rooms. For example - “The bedroom is adjacent to hall and there is a bed in the centre.” This sentence describes bedroom and as well as define an adjacency relation between hall and bedroom.

4. **Information extraction** – From the previous step, we get sentences labeled with room tag or relation tag. We treat setences with relation tag separately to extract relational information. As part of relational information, we extract only door connectivity relation between rooms for now.

   From sentences labeled with room tag, we extract information like shape, dimensions, architectural objects inside room, door connectivity and wall sharing for each tag using techniques of Natural Language Processing like – tokenization, regular expression matching. For extracting rooms and architectural objects, we have built our own custom dictionaries. From the extracted information, we are generating a **JSON** object for each room which contains the following details -

   Room {
   "type" : "hall"
   "shape" : "rectangle"
   "sides" : 4
   "dimensions" : [250, 200, 250, 200]
   "door_placement" : [(2:1), (3:1)]
   "furnitures" : [("sofa":1), ("chair":1)]
   }

   From sentences labeled with relation tag, we make a graph with rooms as nodes and an edge is drawn between two rooms if the rooms are connected by a door. This graph is termed as **Door Connectivity Graph.**

   Thus, this step gives two outputs -
   - List of JSON objects
   - Door Connectivity Graph

5. **Image Rendering** – Using the above extracted information, we generate the output image of floor plan. We traverse the door connectivity graph in Depth First Search fashion and using extracted information of each room and keeping track of previously visited room in recursion function, we keep generating the coordinates of each room and architectual objects and draw the floor plan image. We used SESYD dataset for images of Architectural objects.

Results:

1. Link to the dataset generated after preprocessing - https://drive.google.com/open?id=1RLecrUwFrcfrC_e1rKMODANDU_Jqw4RO

   This link also contains the raw data collected through survey. Survey was done using floor plan images from [5]. No guidelines were given and thus, the description is completely in layman language. By carefully observing the data, it can be concluded that the descriptions are too varied.

2. Text Summarization – It takes input single or multiple descriptions of a floor plan and generate its summary by extractive technique. We implemented text summarization technique using [2].

The floor plan has 5 rooms - kitchen, bathroom, hall, bedroom and dining. Lobby leads to hall. Hall and kitchen are adjacent and connected. Hall has a sofa and a chair. Kitchen leads to dining hall from its right door. Dining area has a table in corner. Dining and bedroom are adjacent. Bathroom and bedroom are adjacent and connected. There is a big bed in bedroom. Bathoom has a tub in corner. There is only a sink in kitchen. Lobby has 2 chairs.

As we enter the house, there is a lobby and it is square shaped with 150 feet. It is at entrance and is a small room. Two chair are there in lobby and there is door on the second wall. The door of lobby leads to hall. Hall is 4 sided with dimensions 250x200 feet and has doors on second and third walls. It has a chair and sofa. First door of hall leads to bedroom. Bedroom is 4 sided with dimensions 200x100 feet and has a door on first wall. It has a bed in it. Door of bedroom leads to bathroom. Bathroom is rectangular with dimensions 100x100 feet. It has a tub inside it. Second door of hall leads to kitchen. Kitchen is 4 sided with dimensions 100x200 feet with a door on second wall. There is a sink and basin inside kitchen. Door of kitchen leads to dining area. Dining area has 4 walls with dimensions 150x100 feet. There is a dining table placed in center.

↓ Reduction Ratio – 0.4

The door of lobby leads to hall. Hall is 4 sided with dimensions 250x200 feet and has doors on second and third walls. First door of hall leads to bedroom. Bedroom is 4 sided with dimensions 200x100 feet and has a door on first wall. Door of bedroom leads to bathroom. Second door of hall leads to kitchen. Kitchen is 4 sided with dimensions 100x200 feet with a door on second wall. Door of kitchen leads to dining area. Dining area has 4 walls with dimensions 150x100 feet. The floor plan has 5 rooms - kitchen, bathroom, hall, bedroom and dining. Hall and kitchen are adjacent and connected. Kitchen leads to dining hall from its right door.

3. Sentence Classification – It takes input the list of sentences and gives tagged sentences as output. We implemented classifier using [3] with an **accuracy of 96%**. For example -

   - There is a refrigerator in the right corner of the room. : “**Kitchen”**
   - The oven is next to a cabinet that holds coffee cups. : “**Kitchen”**
   - It has a small table nearby my bed. : “**Bedroom”**
   - It has both shower and bathtub. : “**Bathroom”**
   - The door of lobby leads to hall. : **“Relation”**
   - The size of bedroom is twice the size of bathroom. : **“Relation”**

4. Information Extraction – It takes tagged sentences as input and generate the following two outputs. The numbering of wall in door placement attribute of room is done in clockwise order.

   - Storing extracted information

Hall is **rectangular** with dimensions **250 x 200** feet. It has doors on **second** and **third** walls. It has a **sofa** and a **chair** placed inside it.

→

Type - Hall
Shape - “”
Sides - 4
Dimensions - [250, 200, 250, 200]
Door_Placement - [(2:1), (3:1)]
Furnitures - [(“sofa”:1), (“chair”:1)]

- Similarly, we extract information for every room.

Type - Lobby
Shape - square
Sides - 4
Dimensions - [150, 150, 150, 150]
Door_Placement - [(2:1)]
Furnitures - [(“chair”:2)]

Type - Hall
Shape - Rectangle
Sides - 4
Dimensions - [250, 200, 250, 200]
Door_Placement - [(2:1), (3:1)]
Furnitures - [(“sofa”:1), (“chair”:1)]

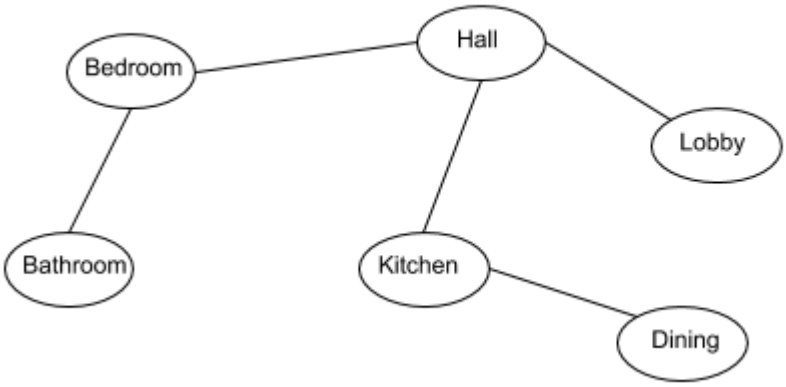
Type - Kitchen
Shape - “”
Sides - 4
Dimensions - [100, 200, 100, 200]
Door_Placement - [(2:1)]
Furnitures - [(“sink”:1), (“basin”:1)]

Type - Dining
Shape - “”
Sides - 4
Dimensions - [150, 100, 150, 100]
Door_Placement - [ ]
Furnitures - [(“table”:1)]

Type - Bedroom
Shape - “”
Sides - 4
Dimensions - [200, 100, 200, 100]
Door_Placement - [(1:1)]
Furnitures - [(“bed”:1)]

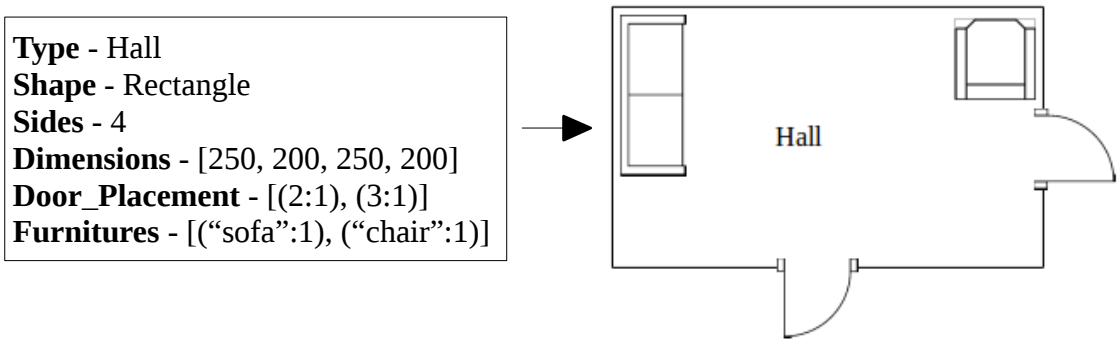
Type - Bathroom
Shape - rectangle
Sides - 4
Dimensions - [100, 100, 100, 100]
Door_Placement - [ ]
Furnitures - [(“tub”:1)]

- Door Connectivity Graph

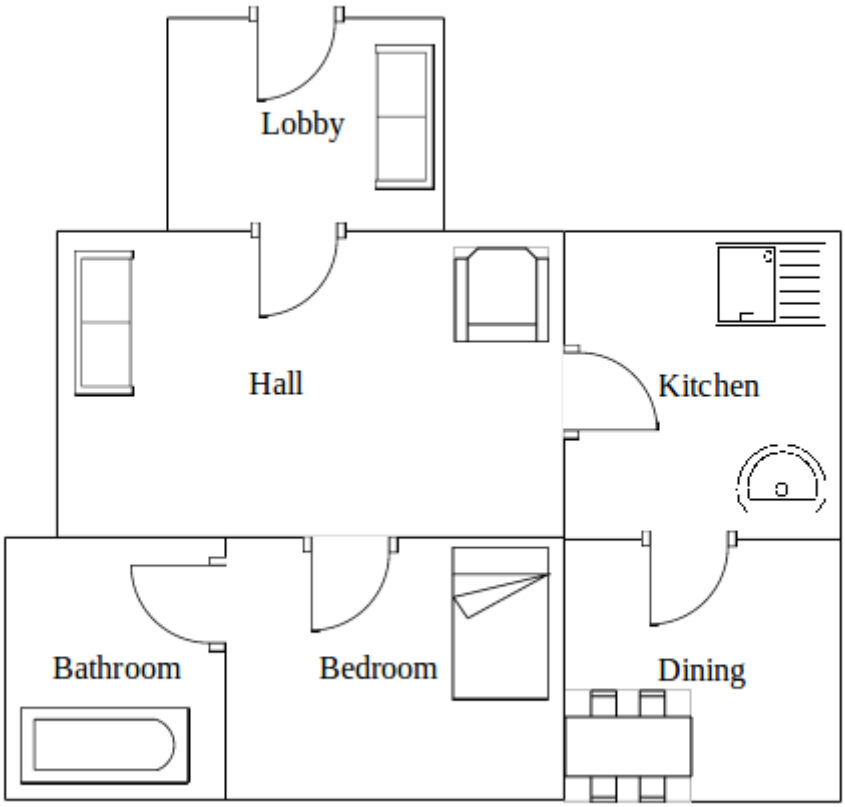


5. Image Rendering – The image generation is done using PIL module of python. The coordinates of room are generated in recursion and random corner is picked to place the architectural objects inside room.

    - Single Room image – It is generated using the extracted information stored in room objects. Images of all the room objects are generated.

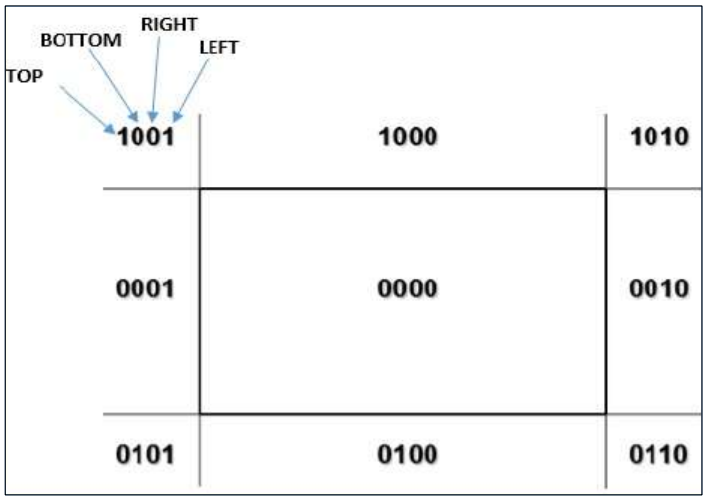


    - Floor plan image – This image is generated by traversing the above door connectivity graph in DFS fashion. We generate the image of each room using the extracted information stored in JSON objects in previous step and then stitch together the individual components during traversal to generate final image.

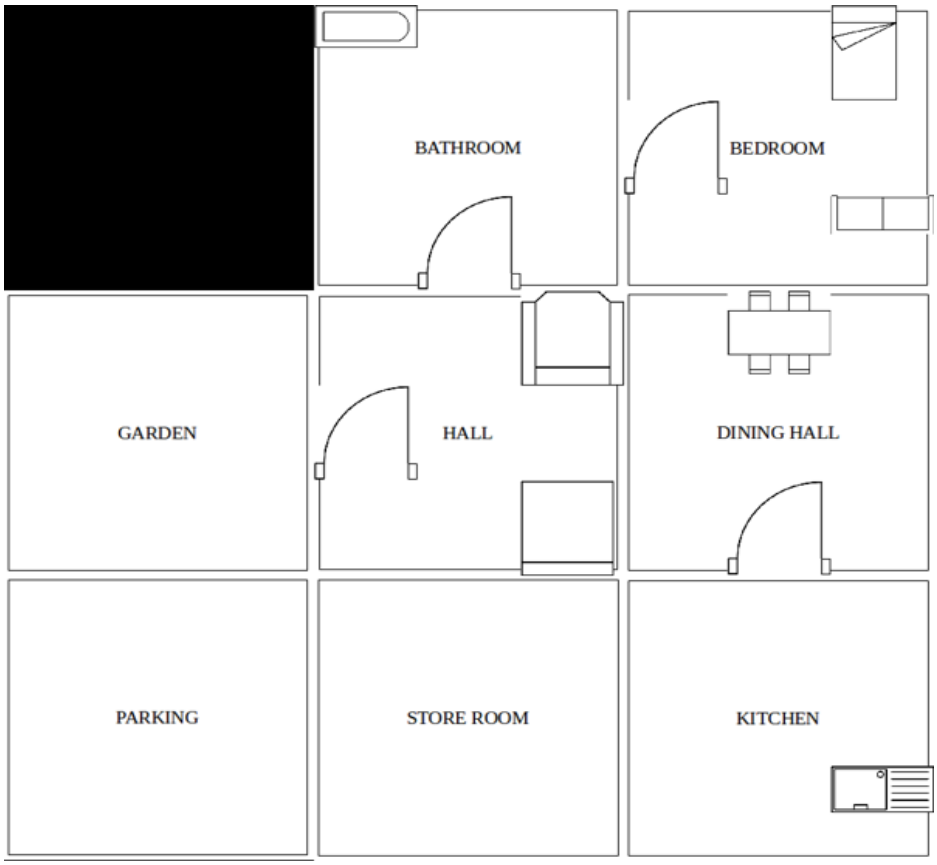


Progress with respect to previous semester: Our previously implemented model was very naive –

- It divided the image into 9 parts (TBRL placement model) and map the rooms to one of the regions.



- The directions were described as north eastern or top right (1010), central (0000), south western or bottom left (0101) etc.
- The global locations of every room and architectural entities were required be mentioned in the description.
- Every room was considered to be of same size as dimensions were not dealt in such model.
- The output from our previous model was as follows -



Our current model is very flexible as compared to model implemented in previous semester.

- We generated our own dataset through survey.
- The placement model used in our current approach is capable of handling arbitrary shaped floor plans.
- We have listed out the required attributes of room which are necessary to be extracted for our model.
- We are able to establish relational graphs among entities.
- We are able to generate an end to end model for image generation from text.

Minor Constraints:

- There are constraints in preprocessing step. For example – Sometimes, the usage of pronouns make the description ambiguous and its is difficult to determine what it refers to. Replacing pronouns with just previously used nouns is a possible solution to this problem but it cannot handle all such cases.
- In classification step, it is difficult to classify some generic sentences which can belong to any room. For example - “The dimensions of the room is 200x300 feet.”

Conclusion: Implemented an end to end model for generation of floor plan image given its textual description taking reasonable assumptions and with good accuracy.

Future Scope:

- The same model can be extendend to deal with multiple rooms of same type(bedroom, bathroom etc.).
- This model deals with only 4-sided rooms for now, but can be easily modified to deal with arbitrary shaped rooms in future.
- In the current model, placement position of furniture with respect to room is not extracted. The arcitectural objects are placed in room randomly. Our model can be easily extendend to incorporate such changes.

References:

[1] Python Text Processing with NLTK 2.0 Cookbook, Book by Jacob Perkins

[2] https://github.com/adamfabish/Reduction

[3] https://github.com/dennybritz/cnn-text-classification-tf

[4] Vishal Gupta, Gurpreet Singh Lehal, A survey of Text Summarization Extractive Techniques, Journal of Emerging Technologies in web Intelligence, August 2010

[5] https://github.com/gesstalt/ROBIN4

[6] http://mathieu.delalandre.free.fr/projects/sesyd/symbols/sketches.html

Date: 24/04/18

Mentor’s signature