Collecting Spatial Information for Locations in a Text-to-Scene Conversion System

Masoud Rouhizadeh* Richard Sproat* Bob Coyne**

*Center for Spoken Language Understanding, Oregon Health and Science University
**Department of Computer Science, Columbia University

1 Introduction

WordsEye [1, 2] is a text-to-scene conversion system that receives a text description of a picture from the user via its online interface and converts it into a 3D scene. The core of WordsEye is VigNet, a unified knowledge base and representational system for expressing lexical and real-world knowledge needed to depict scenes from text [3]. In particular, VigNet contains the knowledge needed to map the objects and locations specified in a text into the actual 3D objects. Individual objects (e.g. a chair) typically correspond to single 3D models, but locations (e.g. a living room) are typically composed of several individual objects, and those objects have a typical spatial relation. Prototypical mappings from locations to objects and the spatial relations of those objects are called **location vignettes** [4, 5].

Existing lexical and common sense knowledge resources such as WordNet [6], FrameNet [7], and OpenMind [8] do not contain the spatial and semantic information required to construct location vignettes (for a discussion see [4]), so we need to build our own lexical resource. One of the well-known approaches for building lexical resources is extracting lexical relations from large text corpora. Directly relevant to this paper is a work by Sproat [9], which attempts to extract canonical locations of actions from text corpora. This approach provides useful information, but the extracted data is noisy and requires hand editing. Furthermore, much of the information that we are looking for is common-sense knowledge that is taken for granted by human beings and is not explicitly stated in corpora. Although structured corpora like Wikipedia do mention associated objects, they are often incomplete [4]. In this paper we investigate using Amazon Mechanical Turk (AMT) for building our own domain specific corpus and locations vignettes.

2 Using AMT to build location vignettes

AMT is an online marketplace to co-ordinate the use of human intelligence to perform small tasks such as image annotation that are difficult for computers but easy for humans [10]. The inputs to our AMT tasks are typical photos of different rooms, that show large objects typical of that particular room. We carefully selected the picture from the results of image searches using Google and Bing. Turkers of each task had to be in the US and had previous approval rating of at least 99%. Restricting the location of the Turkers increases the chance that they are native speakers of English, or at least have good command of the language.

Phase 1: Collecting the functionally and visually important objects of rooms

The functionally important objects for a room are those that are required in order for the room to be recognized or to function properly. The visually important objects are those that help define the basic structural makeup of that particular room instance, such as large objects and those that are fixed in location. Examples of those objects in a kitchen can be “stove”, “oven”, “sink”, “cabinets”, and so on. After collecting the objects from several AMT tasks, we post-process them by the following steps [4]:

1. Manual checking of spelling and converting plural nouns to singular.
2. Removing conjunctions such as “and”,“or”, and “/”.
3. Substituting the objects belonging to the same WordNet synset with the most frequent word of the synset. (“tub”, “bath”, and “bathtub” ⇒ “bathtub”)
4. Substituting words with major substrings in common (“night stand”, “night-stand” ⇒ “nightstand”).
5. Selecting the head nouns of compounds (“computer monitor” ⇒ “monitor”).

Phase 2: Collecting the visual properties of the rooms

Turkers should determine the room layout (diagonal or horizontal), room size (small, medium, or large), ceiling height, wall texture (painted color, wallpaper pattern, fabric, wood paneling, tile, concrete, or stone), and floor texture (tile, wood, carpeted, stone, or concrete).

Phase 3: Collecting the spatial relations between the objects

For each object **O** that is collected in phase 1 Turkers should answer the following questions:

1. Is **O** located against a wall? If so, determine the wall.
2. Is **O** near another object? If so, determine the object, determine the direction (front, back, or side), and the distance (1 ft, 2 ft, 3 ft, or 4 ft or more).
3. Is **O** supported by (i.e. on, part of, or attached-to) another object? If so, determine the object.
4. Is **O** facing another object? (e.g. chair facing a table) If so, determine the object.

Building low-level location description corpus

Prior to the above phases, we ask Turkers to provide simple and clear descriptions of the pictured room. We explicitly instruct Turkers that their descriptions have to be in the form of naming the main elements or objects in the room and their positions in relation to each other. To extract location information from the corpus, we process the text with the NLP module of WordsEye system. We extract the objects and other elements of locations that are in the form of RELATION–GROUND–FIGURE (for more details see [4] and [5]).

3 Results

We have completed phases 1 and 2 for 85 rooms and are now performing phase 3 for those rooms. We have also built a low-level description corpus for the 85 rooms that contains around 11,000 words. To evaluate the results of phase 1, we compare the objects we collect to a gold-standard set of objects that are found in five rooms compiled by an expert. 91% of our collected objects were correct (precision) and we could gather 88% of the objects that we expected (recall). We also attainted 67% precision and 79% recall by extracting the objects from the description corpus. (see [4] and [5] for our methodology of extracting objects). We are now investigating methods such as inter-annotator agreement measures for evaluating phases 2 and 3. We also plan to use WordsEye to generate pictures of the constructed location vignettes and have people evaluate how much the generated pictures resemble the typical locations in the real world.

Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No.IIS-0904361.

References

[1] B. Coyne, O. Rambow, J. Hirschberg, and R. Sproat, "Frame semantics in text-to-scene generation," in *Knowledge-Based and Intelligent Information and Engineering Systems*, ser. Lecture Notes in Computer Science, R. Setchi, I. Jordanov, R. Howlett, and L. Jain, Eds. Springer Berlin / Heidelberg, 2010, vol. 6279, pp. 375–384.

[2] B. Coyne and R. Sproat, "Wordseye: An automatic text-to-scene conversion system," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, Los Angeles, CA, 2001, pp. 487– 496.

[3] B. Coyne, D. Bauer, and O. Rambow, "VigNet: Grounding language in graphics using frame semantics," in *ACL Workshop on Relational Models of Semantics (RELMS)*, 2011.

[4] M. Rouhizadeh, D. Bauer, B. Coyne, O. Rambow, and R. Sproat, "Collecting spatial information for locations in a text-to-scene conversion system," in *CoSLI-2: Computational Models for Spatial Languages) at CogSciComputational Models for Spatial Languages*, 2011.

[5] M. Rouhizadeh, B. Coyne, and R. Sproat, "Collecting semantic information for locations in the scenario-based lexical knowledge resource of a text-to-scene conversion system," in *Knowledge-Based and Intelligent Information and Engineering Systems - 15th International Conference, KES 2011, Kaiserslautern, Germany, Proceedings, Part IV*, ser. Lecture Notes in Computer Science, vol. 6884. Springer, 2011, pp. 378–387.

[6] C. Fellbaum, *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA, 1998.

[7] C. Baker, C. Fillmore, and J. Lowe, "The Berkeley Framenet Project," in *Proceedings of the 17th international conference on Computational linguistics*, 1998, pp. 86–90.

[8] P. Singh, T. Lin, E. Mueller, G. Lim, T. Perkins, and W. Li Zhu, "Open mind common sense: Knowledge acquisition from the general public," in *On the Move to Meaningful Internet Systems 2002: CoopIS, DOA, and ODBASE*, ser. Lecture Notes in Computer Science, R. Meersman and Z. Tari, Eds. Springer Berlin / Heidelberg, 2002, vol. 2519, pp. 1223–1237.

[9] R. Sproat, "Inferring the environment in a text-to-scene conversion system," in *Proceedings of The First International Conference on Knowledge Capture*, Victoria, BC, Canada, 2001, pp. 147–154.

[10] C. Callison-Burch and M. Dredze, "Creating speech and language data with amazon's mechanical turk," in *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, Los Angeles, CA, 2010, pp. 1–12.