Frame-Based Representation of Lexical, Graphical, and Factual Knowledge for Text-to-Scene Generation

DANIEL BAUER, BOB COYNE & OWEN RAMBOW
(Columbia University, New York)

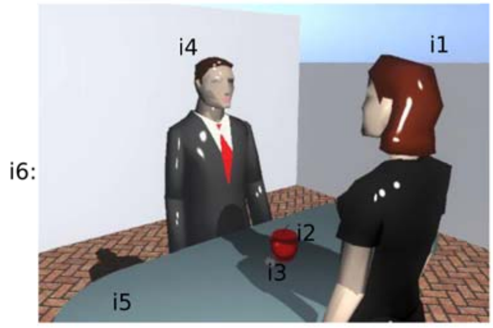
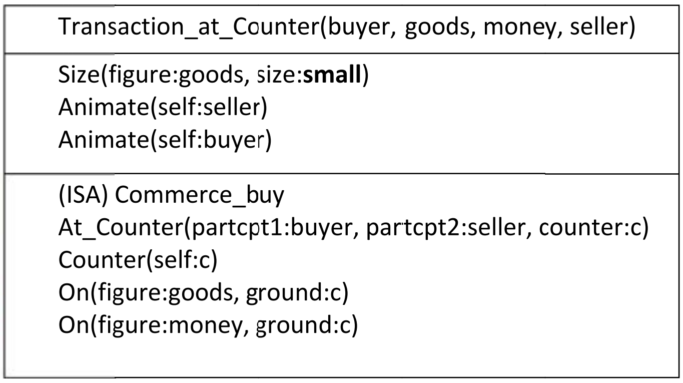


Figure 1 - a possible realization for sentence (1).



Transaction_at_Counter(buyer, goods, money, seller)

Size(figure:goods, size:**small**)
Animate(self:seller)
Animate(self:buyer)

(ISA) Commerce_buy
At_Counter(partcpt1:buyer, partcpt2:seller, counter:c)
Counter(self:c)
On(figure:goods, ground:c)
On(figure:money, ground:c)

Figure 2 – decomposition of one specific event vignette that can realize Commerce_buy and decomposes to the spatial arrangement in Figure 1.

We outline ongoing work on WordsEye, a text-to-scene generation system. While WordsEye (Coyne and Sproat, 2001) currently recognizes descriptions of simple spatial relations between objects, we are aiming to add support for complex actions, events and states-of-affairs. To this end we use frame-based representations for both asserted and terminological knowledge. The resulting scenes are static spatial arrangements of pre-existing 3D models. For instance, the following sentence might produce the scene in Figure 1:

(1) Mary bought an apple for $1

To convert a description into a scene, different sources of knowledge are required: lexical knowledge, graphical knowledge, and world knowledge. We discuss them in turn.

First, lexical knowledge links words in their syntactic context to semantic representations. Such knowledge is already described by frame semantics (Fillmore, 1982) and recorded in FrameNet (Fillmore et al, 2003 and Ruppenhofer, 2010), which we build on. To analyze sentence (1), a semantic parser would label *buy* with the **Commerce_buy** frame and assign the frame elements **buyer** to *Mary*, **goods** to *an apple*, and **money** to *for $1*. In FrameNet annotations, frame elements are filled with text spans. In contrast, we create a single semantic representation for the whole sentence: we fill frame elements recursively with further frame instances. Each frame instance is associated with its type (the frame). Frame instances, shown as $i_1 \dots i_6$ in Figure 1, are the entities or events mentioned in the description.

Second, we need graphical knowledge about the arrangements of 3D models in a scene. FrameNet frames describe functional relations between frame elements, without characterizing the nature of the relation in detail. As we are interested in generating static 3D scenes, we require knowledge about the spatial relations between actual entities needed for visualization. For this purpose, we extend FrameNet frames by adding specific visually-oriented information. We call these extended frames *vignettes*. In order to represent scenes, vignettes a) optionally introduce new frame elements representing additional entities required to convey the manner in which an action is carried out; b) limit certain frame elements to certain classes of fillers, such as small round objects; and c) specify concrete 3D models (most entity frames) or a set of sub-frames representing graphical relations between entities participating in the frame (event frames). Figure 2 shows the decomposition of one of several possible vignettes extending **Commerce_buy**. **On** is a primitive graphical frame that can be interpreted directly by low-level spatial inference. **At_counter** describes a common template for scenes in which two parties interact over a counter. Vignettes are connected to their lexical super-frame via inheritance and selected by selectional restrictions on the frame elements.

Apple()

ISA(Fruit)
Size(fig:self, size:**small**)
Shape(fig:self, shape:**round**)

Figure 3 - a basic vignette for apples.

Finally, we need to represent factual knowledge about objects as well as selectional restrictions. We use FrameNet's inheritance frame-to-frame relation to build an ontology of concepts. To assert selectional restrictions and properties of objects, all frames carry a **self** frame element, relating to the frame instance itself, which allows us to define properties of a Frame or Vignette. Figure 3 shows a simple definition for **Apple** that works with the above definition for **Commerce_counter**.

Our semantic resource, called *VigNet* (Coyne et al, 2011), currently contains all of FrameNet and about 3000 3D models, with information about their properties (size, color, shape, texture). VigNet also contains a number of handcrafted abstract vignettes (similar to **At_counter** above) for situations and events, as well as rooms. New vignettes are being added using Amazon Mechanical Turk and the WordsEye system itself.

To create a scene from an input sentence two inference levels are required: Resolving high-level frame semantics into vignettes and interpreting primitive spatial relations (**On**, **Near**…) using spatial reasoning to create an actual 3D scene. Here we focus only on the first subtask. WordsEye already supports spatial inference and support for more elaborate reasoning (in rooms and other environments) is currently being added.

To convert a high-level FrameNet-style semantic parse into a vignette semantic description, we first analyze the sentence syntactically and create n-best FrameNet-style semantic parses for each frame evoking word and its frame elements in isolation. As other semantic parsers do not support n-best analyses, we developed our own semantic parser. The parser maps a new input parse to frame annotations observed in the FrameNet data using a probabilistic model of alignments between syntactic dependency structures. The set of annotation hypotheses thus derived is ranked using semantic information. Initial results have shown that the gold frame structure is recovered within the best-10 results most of the time (~80%) and we are currently optimizing the ranking model. We can construct a forest of possible analyses for the entire sentence from the n-best annotations for each sentence. From this forest we need to select a single tree that can be rendered into a scene.

Not only does the system need to select appropriate lexical frames for each frame-evoking element in the sentence, it also needs to find suitable vignette extensions for these frames. The difficulty is that all selected vignettes need to be mutually consistent. As frame descriptions do not involve quantification and negation, we only need to check if the selectional restrictions are met. On the other hand, finding a set of consistent vignettes for a sentence is hard. In a first implementation, we will search the entire space of possible assignments for all lexical frames proposed by the parser, but several heuristics can be employed. For instance, we can first assign the most specific vignette to the main frames in a sentence, i.e. the vignette that has the most constraints on its frame elements. This limits the choice for other vignettes. We also expect this strategy to produce more interesting visualizations. In future work we are planning to make factual and graphical knowledge provided by VigNet available to the semantic parser, integrating linguistic semantic analysis and inference more tightly.

Our work stands in a tradition of semantic analysis using decomposition into primitives, for instance Conceptual Dependency theory (Schank and Abelson, 1977), the Generative Lexicon (Pustejovsky, 1991), Event Logic (Siskind, 1994), and VerbNet's (Kipper Schuler, 2005) definitions of verb class semantics. Other related work deals with grounding of semantic representations in graphical relations (Simmons, 1975, Kahn, 1979, Ma and McKevitt, 2003), ontologies (Nirenburg and Raskin, 2004, Scheffczyk et al, 2006, Yu et al, 2007), or perceptual and motoric embodiment (Bergen and Chang, 2005).

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

Bergen, B.K. & Chang, N. (2005). Embodied construction grammar in simulation-based language understanding. In: Construction grammars: Cognitive grounding and theoretical extensions.

Coyne, B & Sproat, R. (2001). WordsEye: an automatic text-to-scene conversion system. Proceedings of SIGGRAPH.

Coyne, B., Bauer, D. & Rambow, O. (2011).VigNet: Grounding Language in Graphics using Frame Semantics. Proceedings of the ACL Workshop on Relational Models of Semantics.

Das, D., Schneider, N., Chen, D. & Smith, N.A. (2010). Probabilistic frame-semantic parsing. Proceedings of NAACL-HLT.

Fillmore, C.J. (1982). Frame Semantics. In: Linguistics in the morning Calm. Hanshin Pub.Co.

Fillmore, C.J, Johnson, C.R. & Petruck, M.R.L. (2003). Background to FrameNet. International Journal of Lexicography 16 (3). 235-250.

Kahn, K. (1979). Creation of Computer Animation from Story Descriptions. Ph.D. thesis, AI Lab, MIT.

Kipper Schuler, K.(2005). VerbNet: A broad-coverage, comprehensive verb lexicon. Ph.D. thesis. University of Pennsylvania.

Ma, M. & McKevitt, P. (2006). Virtual human animation in natural language visualisation. Artificial Intelligence Review. 25. 37–53.

Nirenburg, S. and Raskin, V. (2004). Ontological Semantics. Cambridge: MIT Press.

Pastra, K. (2008). PRAXICON: The Development of a Grounding Resource. Proceedings of the International Workshop on Human-Computer Conversation.

Pustejovsky, J. (1991). The generative lexicon. Computational Linguistics. 17 (4).

Ruppenhofer, J., Ellsworth, M., Petruck, M.R.L., Johnson, C.R. & Scheffczyk, J. (2010). FrameNet II: Extended Theory and Practice. Available online. Berkeley: ICSI. http://framenet.icsi.berkeley.edu/.

Schank, R. C. & Abelson, R. (1977). Scripts, Plans, Goals, and Understanding. Earlbaum.

Simmons, R.. (1975) The CLOWNS Microworld. Proceedings of the Workshop on Theoretical Issues in Natural Language Processing.

Siskind, J.M. (1994). Grounding language in perception. Artificial Intelligence Review. 8 (5).

Yu, L.C., Wu, C.H., Philpot, A. & Hovy, E. (2007). OntoNotes: Sense pool verification using Google N-gram and statistical tests. Proceedings of the ISWC OntoLex Workshop.