Easy as ABC? Facilitating Pictorial Communication via Semantically Enhanced Layout

Andrew B. Goldberg, Xiaojin Zhu, Charles R. Dyer, Mohamed Eldawy, Lijie Heng
Department of Computer Sciences
University of Wisconsin, Madison, WI 53706, USA
{goldberg, jerryzhu, dyer, eldawy, ljheng}@cs.wisc.edu

Abstract

Pictorial communication systems convert natural language text into pictures to assist people with limited literacy. We define a novel and challenging problem: picture layout optimization. Given an input sentence, we seek the optimal way to lay out word icons such that the resulting picture best conveys the meaning of the input sentence. To this end, we propose a family of intuitive "ABC" layouts, which organize icons in three groups. We formalize layout optimization as a sequence labeling problem, employing conditional random fields as our machine learning method. Enabled by novel applications of semantic role labeling and syntactic parsing, our trained model makes layout predictions that agree well with human annotators. In addition, we conduct a user study to compare our ABC layout versus the standard linear layout. The study shows that our semantically enhanced layout is preferred by non-native speakers, suggesting it has the potential to be useful for people with other forms of limited literacy, too.

1 Introduction

A picture is worth a thousand words—especially when you are someone with communicative disorders, a foreign language speaker, or a young child. Pictorial communication systems aim to automatically convert general natural language text into meaningful pictures. A perfect pictorial communication system can turn signs and operation instructions into easy-to-understand graphical forms; combined with optical character recognition input, a personal assistant device could create such visual translations on-the-fly without the help of a caretaker. Pictorial communication may also facilitate literacy development and rapid browsing of documents through pictorial summaries.

Pictorial communication research is in its infancy with a spectrum of experimental systems, which we review in Section 2. At one end of the spectrum, some systems render highly realistic 3D scenes but require specific scene-descriptive language. At the other end, some systems perform dictionary-based iconic transliteration (turning words into icons[1] one by one) on arbitrary text but the pictures can be hard to understand. We are interested in using pictorial communication as an assistive communication tool. Thus, our system needs to be able to handle general text yet produce easy-to-understand pictures, which is in the middle of the spectrum. To this end, our system adopts a "collage" approach (Zhu et al., 2007). Given a piece of text (e.g., a sentence), it first identifies important and easy-to-depict words (or phrases) with natural language processing (NLP) techniques. It then finds one good icon per word, either from a manually created picture-dictionary, or via image analysis on image search results. Finally, it lays out the icons to create the picture. Each step involves several interesting research problems.

This paper focuses exclusively on the picture layout component and addresses the following question: Can we use machine learning and NLP techniques to learn a good picture layout that im-

[1] In this paper, an *icon* refers to a small thumbnail image corresponding to a word or phrase. A *picture* refers to the overall large image corresponding to the whole text.

CoNLL 2008: Proceedings of the 12th Conference on Computational Natural Language Learning, pages 119–126
Manchester, August 2008

proves picture comprehension for our target audiences of limited literacy? We first propose a simple yet novel picture layout scheme called "ABC." Next, we design a Conditional Random Field-based semantic tagger for predicting the ABC layout. Finally, we conduct a user study contrasting our ABC layout to the linear layout used in iconic transliteration. The main contribution of this paper is to introduce the novel task of layout prediction, learned using linguistic features including PropBank role labels, part-of-speech tags, and lexical features.

2 Prior Pictorial Communication Work

At one extreme, there has been significant prior work on "text-to-scene" type systems, which were often intended to aid graphic designers in placing objects in a 3D environment. Example systems include NALIG (Adorni et al., 1983), SPRINT (Yamada et al., 1992), Put (Clay and Wilhelms, 1996), and others (Brown and Chandrasekaran, 1981). Perhaps the best known system of this type, WordsEye (Coyne and Sproat, 2001), uses a large manually tagged collection of 3D polyhedral models to create photo-realistic scenes. Similarly, CarSim (Johansson et al., 2005) can create animated scenes, but operates exclusively in the limited domain of reconstructing road accidents from traffic reports. These systems cater to detailed descriptive text with visual and spatial elements. They are not intended as assistive tools to communicate general text, which is our goal.

Several systems (Zhu et al., 2007; Mihalcea and Leong, 2006; Joshi et al., 2006) attempt to balance language coverage versus picture sophistication. They perform some form of keyword selection, and select corresponding icons automatically from a 2D image database. The result is a pictorial summary representing the main idea of the original text, but precisely determining the original text by looking at the picture can be difficult.

At the other extreme, augmentative and alternative communication software allows users to input arbitrary text. The words, and sometimes common phrases, are semi-automatically transliterated into icons, and displayed in sequential order. Users must learn special icons, which correspond to function words, before the resulting pictures can be fully understood. Examples include SymWriter (Widgit Software, 2007) and Blissymbols (Hehner, 1980).

Other than explicit scene-descriptive languages, pictorial communication systems have not sufficiently addressed the issue of picture layout for general text. We believe a good layout can better communicate the text a picture is trying to convey. The present work studies the use of a semantically inspired layout to enhance pictorial communication. For simplicity, we restrict our attention to the layout of a single sentence. We anticipate the use of text simplification (Chandrasekar et al., 1996; Vickrey and Koller, 2008) to convert complex text into a set of appropriate inputs for our system.

3 The ABC Layout

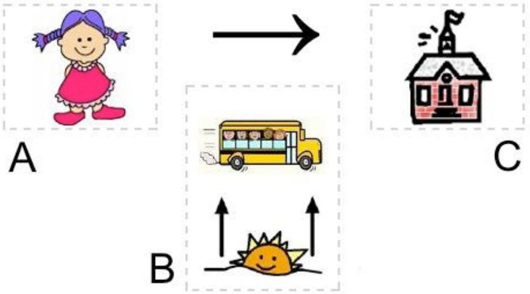
A good picture layout scheme must be intuitive to humans and easy to generate by computers. To design such a layout, we conducted a pilot study. Five human annotators produced free-hand pictures of many sentences. Analyzing these pictures, we found a large amount of agreement in the use of arrows to mark actions and to provide structure to what would otherwise be a jumble of icons.

Motivated by the pilot study, we propose a simple layout scheme called ABC. It features three *positions*, referred to as A, B, and C. In addition, an arrow points from A through B to C (Figure 1). These positions are meant to denote certain semantic roles: roughly speaking, A denotes "who," B denotes "what action," and C denotes "to whom, for what." Each position can contain any number of icons, each representing a word or phrase in the text. Words that do not play a significant role in the text will be omitted from the ABC layout.

There are two main advantages of the ABC layout:

1. The ABC positioning of icons allows users to infer the semantic role of the corresponding concepts. In particular, we found that verbs can be difficult to depict and understand without such hints. The B position serves as an action indicator to disambiguate between multiple senses of the same icon. For example, in Figure 1, the school bus icon clearly represents the verb phrase "rides the bus," rather than just the noun "bus."

2. Such a layout is particularly amenable to machine learning. Specifically, we can turn the problem of finding the optimal layout for an input sentence into a sequence tagging problem, which is well-studied in NLP.



The girl rides the bus to school in the morning
O A B B B O C O O B

Figure 1: Example ABC picture layout, original text, and tag sequence.

3.1 ABC Layout as Sequence Tagging

Given an input sentence, one can assign each word a tag from the set $\{$A, B, C, O$\}$. The bottom row in Figure 1 shows an example tag sequence. The tag specifies the ABC layout position of the icon corresponding to that word. Tag O means "other" and marks words not included in the picture. Within each position, icons appear in the word order in the input sentence. Therefore, a tag sequence uniquely determines an ABC layout of the picture.

Finding the optimal ABC layout of the input sentence is thus equivalent to computing the most likely tag sequence given the input sentence. We adopt a machine learning approach by training a sequence tagger for this task. To do so, we need to collect labeled training data in the form of sentences with manually annotated tag sequences. We discuss our annotation effort next, and present our machine learning models in Section 4.

3.2 Human Annotated Training Data

We asked the five annotators to manually label 571 sentences compiled from several online sources, including grade school texts about history and science, children's books, and recent news headlines. Some sentences were written by the annotators and describe daily activities. The annotators tagged each sentence using a Web-based tool to drag-and-drop icons into the desired positions in the layout[2].

To gauge the quality of the manually labeled data, and to understand the difficulty of the ABC layout, we computed inter annotator agreement among three of the five annotators on a common set of 48 sentences. Considering all pair-wise comparisons of the three annotators, the overall average tag agreement was 77%. This measures the total number of matching tags (across all sentences) divided by the total number of tags. Matching strictly requires both the correct tag and the correct modifier. We also computed Fleiss' kappa, which measures the degree of inter-annotator agreement beyond the amount expected by chance (Fleiss, 1971). The values range from 0 to 1, with 1 indicating perfect agreement. The kappa statistic was 0.71, which is often considered moderate to high agreement.

Further inspection revealed that most disagreement was due to annotators reversing A and C tags. This could arise from interpreting passive sentences in different ways or trying to represent physical movement. For example, some annotators found it more natural to depict eating by placing a food item in A and the eater in C, treating the arrow as the transfer of food. It was also common for annotators to disagree on whether certain adverbs and time modifiers belong in B or in C. These differences all suggest the highly subjective nature of conceptualizing pictures from text.

4 A Conditional Random Field Model for ABC Layout Prediction

We now introduce our approach to automatically predicting the ABC layout of an input sentence. While it was most natural for human annotators to annotate text at the word level, early experiments quickly revealed that predicting tags at this level is quite challenging. Most of this stems from the fact that human annotators tend to fragment the text into many small segments based on the availability of good icons. For example, the phrase "the white pygmy elephant" may be tagged as "O A O A" because it is difficult for the annotator to find an icon of this exact phrase or the word "pygmy," but easy to find icons of "white" and "elephant" separately. Essentially, human annotation combines two tasks in one: deciding where each phrase goes in the layout, and deciding which words within a phrase can be depicted with icons.

To rectify this situation, we make layout predictions at the level of chunks (phrases); that is, we automatically break the text into chunks, then predict one A, B, C, or O tag for each chunk. Since the

[2] The manual tagging actually employs a more detailed tag set to denote phrase structure: Each A, B, or C tag is combined with a modifier of b (begin phrase) or i (inside phrase). For example, the phrase "rides the bus" in Figure 1 is tagged with B_b B_i B_i, and shares one icon. The icons were also manually selected by the annotator from a list of Web image search results.

tag choices made for different chunks may depend on each other, we employ Conditional Random Fields (CRF) (Lafferty et al., 2001), which are frequently used in sequential labeling tasks like information extraction. Our choice of chunking is described in Section 4.1, and the CRF models and input features are described in Section 4.2. The task of deciding which words within a chunk should appear in the picture is addressed by a "word picturability" model, and is discussed in a separate paper.

For training, we automatically map the word-level tags in our annotated data to chunk-level tags based on the majority ABC tag within a chunk.

4.1 Chunking by Semantic Role Labeling

Ideally, we would like semantically coherent text chunks to be represented pictorially in the same layout position. To obtain such chunks, we leverage existing semantic role labeling (SRL) technology (Palmer et al., 2005; Gildea and Jurafsky, 2002). SRL is an active NLP task in which words or phrases in a sentence are assigned a label indicating the role they play with respect to a particular verb (also known as the target predicate). SRL systems like FrameNet (Baker et al., 1998) and PropBank (Palmer et al., 2005) aim to provide a rich representation for applications requiring some degree of natural language understanding, and are thus perfectly suited for our needs. We shall focus on PropBank labels because they are easier to use for our task. To obtain semantic role labels, we use the automatic statistical semantic role labeler ASSERT (Pradhan et al., 2004), trained to identify PropBank arguments through the use of support vector machines and full syntactic parses.

To understand how SRL can be useful for deriving pictorial layouts, consider the sentence "The boy gave the ball to the girl." PropBank marks the semantic role labels of the "arguments" of verbs. The target verb "give" is part of the frameset "transfer," with core arguments "Arg0: giver" (the boy), "Arg1: thing given" (the ball), and "Arg2: entity given to" (the girl). Verbs can also involve non-core modifier arguments, such as ArgM-TMP (time), ArgM-LOC (location), ArgM-CAU (cause), etc. The entities playing semantic roles are likely to be entities we want to portray in a picture. For PropBank, Arg0 often represents an Agent, and Arg1 the Patient or Theme. If we could map the different semantic role labels to ABC tags with simple rules, then we would be done.

Unfortunately, it is not this simple, as PropBank roles are verb-specific. As Palmer et al. pointed out, "No consistent generalizations can be made across verbs for the higher-numbered arguments" (Palmer et al., 2005). In the above example, we might expect a layout rule of [Arg0]→A, [Target, Arg1]→B, [Arg2]→C. However, this rule does not generalize to other verbs, such as "drive," as in the sentence "The boy drives his parents crazy," which also has three core arguments "Arg0: driver," "Arg1: thing in motion," and "Arg2: secondary predication on Arg1." However, here the action is figurative, and we would expect a layout rule that puts Arg1 in position C: [Arg0]→A, [Target]→B, [Arg1,Arg2]→C.

In addition, while modifier arguments have the same meaning across verbs, their pictorial representation may differ based on context. Consider the sentences "Polar bears live in the Arctic." and "Yesterday at the zoo, the students saw a polar bear." In the former, a human annotator is likely to place an icon for the ArgM-LOC "in the Arctic" in position C (e.g., following a polar bear icon in A and a house icon in B). However, the ArgM-LOC in the second sentence, "at the zoo," seems more appropriately placed in position B since it describes where this particular action occurred.

Finally, the situation is further complicated when a sentence contains multiple verbs. SRL treats each verb in isolation, producing multiple sets of role labels, yet our goal is to produce a single picture. Clearly, the mapping from semantic roles to layout positions is non-trivial. We describe our statistical machine learning approach next.

4.2 Our CRF Models and Features

We use a linear-chain CRF as our sequence tagging model. A CRF is a discriminative model of the conditional probability $p(\mathbf{y}|\mathbf{x})$, where $\mathbf{y}$ is the sequence of layout tags in $\mathcal{Y}=\{$A,B,C,O$\}$, and $\mathbf{x}$ is the sequence of SRL chunks produced by the process described in Section 4.1. Our CRF has the general form

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp\left(\sum_{t=1}^{|\mathbf{x}|}\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}, t)\right)$$

where the model parameters are $\{\lambda_k\}$. We use binary features $f_k(y_t, y_{t-1}, \mathbf{x}, t)$ detailed below. Finally, we use an isotropic Gaussian prior $N(0, \sigma^2 I)$ on parameters as regularization.

We explored three versions of the above model by specializing the weighted feature function $\lambda_k f_k()$. **Model 1** ignores the pairwise label potentials and treats each labeling prediction independently: $\lambda_{jk}\mathbf{1}_{\{y_t=j\}}f_k(\mathbf{x},t)$, where $\mathbf{1}_{\{z\}}$ is an indicator function on z. This is equivalent to a multi-class logistic regression classifier. **Model 2** resembles a Hidden Markov Model (HMM) by factoring pairwise label potentials and emission potentials: $\lambda_{ij}\mathbf{1}_{\{y_{t-1}=i\}}\mathbf{1}_{\{y_t=j\}}+\lambda_{jk}\mathbf{1}_{\{y_t=j\}}f_k(\mathbf{x},t)$. Finally, **Model 3** has the most general linear-chain potential: $\lambda_{ijk}\mathbf{1}_{\{y_{t-1}=i\}}\mathbf{1}_{\{y_t=j\}}f_k(\mathbf{x},t)$. Model 3 is the most flexible, but has the most weights to learn.

We use the following binary predicate features $f_k(\mathbf{x},t)$ in all our models, evaluated on each chunk produced by the semantic role labeler:

1. PropBank role label(s) of the chunk (e.g., Target, Arg0, Arg1, ArgM-LOC). A chunk can have multiple role labels if the sentence contains multiple verbs; in this case, we merge the multiple SRL results by taking their union.

2. Part-of-speech tags of all the words in the chunk. All syntactic parsing results are obtained from the Stanford Parser (Klein and Manning, 2003), using the default PCFG model.

3. Phrase type (e.g., NP, VP, PP) of the deepest syntactic parse tree node covering the entire chunk. We also include a feature indicating whether the phrase is nested within an ancestor VP.

4. Lexical features: individual word identities in the top 5000 most frequent words in the Google 1T 5gram corpus (Brants and Franz, 2006). For other words, we use their automatically predicted WordNet supersenses (Ciaramita and Altun, 2006). Supersenses are 41 broad semantic categories (e.g., noun.location, verb.communication). By dividing lexical features in this way, we hope to learn specific qualities of common words, but generalize across rarer words.

We also experimented with features derived from typed dependency relations, but these did not improve our models. We suspect the PropBank role labels capture much of the same information. In addition, the Google 5000-word list was the best among several word lists that we explored for splitting up the lexical features.

4.3 CRF Experimental Results

We trained our CRF models using the MALLET toolkit (McCallum, 2002). Our complete dataset consists of the 571 manually annotated sentences (tags mapped to chunk-level). The only tuning parameter is the Gaussian prior variance, σ^2. We performed 5-fold cross validation, varying σ^2 and comparing performance across models. Figure 2 demonstrates that peak per-chunk accuracy (77.6%) and macro-averaged F1 scores are achieved using the most general sequence labeling model. As a result, the user study in the next section is based on layouts predicted by Model 3 with $\sigma^2 = 1.0$, trained on all the data.

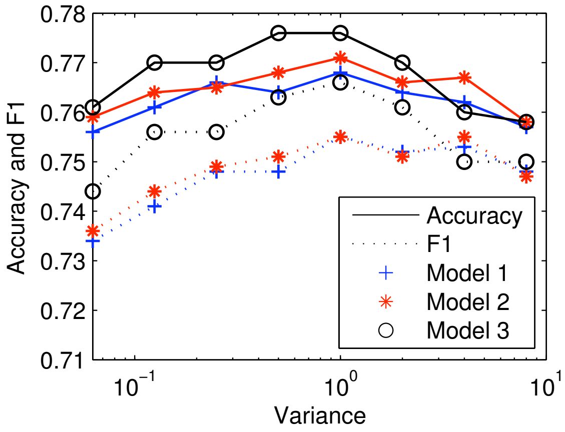


Figure 2: 5-fold cross validation results for different values of the regularization parameter (variance σ^2) and three CRF models predicting A, B, C, or O layout tags.

To understand which features contribute most to performance, we experimented with removing each of the four types (individually). Peak accuracy drops the most when lexical features are removed (76.4%), followed by PropBank features (76.5%), phrase features (76.9%), and POS features (77.1%).

The features in the final learned model make intuitive sense. It prefers tag transitions A→B and B→C, but not A→C or C→A. The model likes the word "I" and noun phrases (not nested in a verb phrase) to have tag A. Verbs and ArgM-NEGs are frequently tagged B, while noun.object's, Arg4s, and ArgM-CAUs are typically C. The model discourages Arg0s and conjunctions in B, and dislikes adverbial phrases and noun.time's in C.

While 77.6% cross validation accuracy may seem low, it is in fact close to the 81% inter annotator agreement[3], and thus close to optimal. The confusion matrix (not shown) reveals that most er-

[3] The 81% agreement is on mapped chunk-level tags without modifiers (Fleiss' kappa 0.74), while the 77% agreement in Section 3.2 is on word-level tags with modifiers.

rors probably arise from disagreements in the individual annotators. The most common errors are predicting B for chunks labeled O and confusing tags B and C. Manually inspecting the pictures in our training set shows that annotators often omitted the verb (such as "is" or "has") and left the B position empty, since it could be inferred by the presence of the arrow and the images in A and C. Also, annotators tended to disagree on the location of adverbial expressions, dividing them between positions B and C. Finally, only 3.3% of chunks were incorrectly omitted from the pictures. Therefore, we conclude that our CRF models are capable of predicting the ABC layouts.

5 User Study

We have proposed the ABC layout, and showed that we can learn to predict it reasonably well. But an important question remains: *Can the proposed ABC layout help a target audience of limited literacy understand pictures better, compared to the linear layout used in state-of-the-art augmentative and alternative communication software?* We describe a user study as our first attempt to answer this question. This line of work has two main challenges: one is the practical difficulty of working with human subjects of limited literacy; the other is the lack of a quantitative measure of picture comprehension.

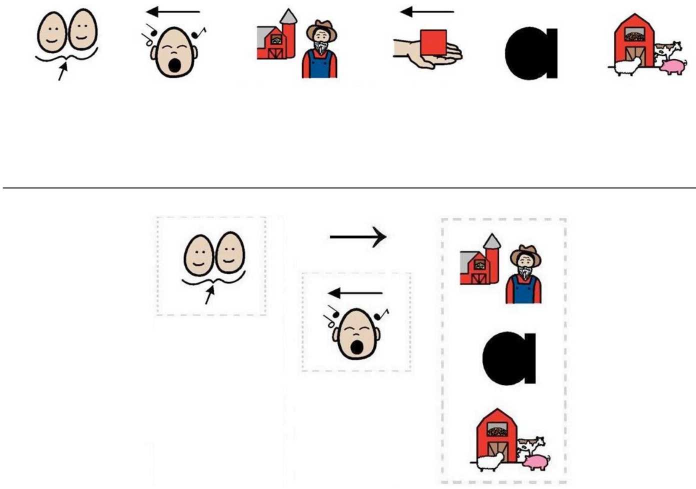
[Subjects]: To partially overcome the first challenge, we recruited two groups of subjects with medium and high literacy respectively, in hopes of extrapolating our findings towards the low literacy group. Specifically, the medium group consisted of seven non-native English speakers who speak some degree of English—"medium literacy" refers to their English fluency; twelve native English speakers comprised the high literacy group. All subjects were adults and did not include the authors of this paper or the five annotators. The subjects had no prior exposure to pictorial communication systems.

[Material]: We randomly chose 90 test sentences from three sources[4] representing our target application domains: short narratives written by and for individuals with communicative disorders (symbolworld.org); one-sentence news synopses written in simple English targeting foreign language learners (simpleenglishnews.com); and the child writing sections of the LUCY corpus (Sampson, 2003). We created two pictures for each test sentence: one using a linear layout and one using an ABC layout. For the linear layout, we used SymWriter. Typing text in SymWriter automatically produces a left-to-right sequence of icons, chosen from an icon database. In cases where SymWriter suggests several possible icons for a word, we manually selected the best one. For words not in the database, we found appropriate thumbnail images using Web image search. This is how a typical user would use SymWriter. To produce the ABC layout, we applied the trained CRF tagger Model 3 to the test sentence. After obtaining A, B, C, and O tags for text chunks, we placed the corresponding icons (from SymWriter's linear layout) in the correct layout positions. Icons for words tagged O did not appear in the ABC version of the picture. Aside from this difference, both pictures of each test sentence contained exactly the same icons—*the only difference was the layout*.

[Protocol]: All 19 subjects observed each of the 90 test sentences exactly once: 45 with the linear layout and 45 with the ABC layout. The layouts and the order of sentences were both randomized throughout the sequence, and the subjects were counter-balanced so each sentence's linear and ABC layouts were viewed by roughly equal numbers of subjects. At the start of the study, each subject read a brief introduction describing the task and saw an example of each layout style. Then for each test sentence, we displayed a picture, and the subject typed a guess of the underlying sentence. Finally, the subject provided a confidence rating (2="almost sure," 1="maybe correct," or 0="no idea"). We measured response time as the time from image display until sentence/rating submission. Figure 3 shows a test sentence in both layouts, together with several subjects' guesses.

[Evaluation metrics]: As noted above, the second main challenge is measuring picture comprehension—we need a way to compare the original sentences with the subjects' guesses. In many ways, this is like machine translation (via pictures), so we turned to two automatic evaluation metrics: BLEU-1 (Papineni et al., 2002) and METEOR (Lavie and Agarwal, 2007). BLEU-1 computes unigram precision (i.e., fraction of response words that exactly match words in the original), multiplied by a brevity penalty for omit-

[4] Distinct from the sources of the 571 training sentences.



"we sing a song about a farm."
"i sing about the farm and animals"
"we sang for the farmer and he gave us animals."
"Someone went to his grandfather's farm
and played with the animals"
"i can't sing in the choir because i have to tend
to the animals."
"twins sing old macdonald has a farm"

"they sang about a farm"
"they sing old mcdonald had a farm."
"we have a farm with a sheep, a pig and a cow."
"two people sing old mcdonald had a farm"
"we sang old mcdonald on the farm."
"they both sing 'old macdonald had a farm'."

Figure 3: The linear and ABC layout pictures for the test sentence "We sang Old MacDonald had a farm." and some subjects' guesses. Note the predicted ABC layout omits the ambiguous "had" icon.

ting words. In contrast, METEOR finds a one-to-one word alignment between the texts that allows partial matches (after stemming and by considering WordNet-based synonyms) and optionally ignores stop words. Based on this alignment, unigram precision, recall, and weighted F measure are computed, and the final METEOR score is obtained by scaling F to account for word-order preservation. We computed METEOR using its default parameters and the stop word list from the Snowball project (Porter, 2001).

[Results]: We report average METEOR and BLEU scores, confidence ratings, and response time for the 4 conditions (native vs. non-native, ABC vs. linear) in Table 1. The most striking observation is that native speakers perform better (in terms of METEOR and BLEU) with the linear layout, while non-native speakers do better with ABC. [5]

	Non-native		Native	
	ABC	Linear	ABC	Linear
METEOR	**0.1975**	0.1800	0.2955	**0.3335**
BLEU	**0.1497**	0.1456	0.2710	**0.3011**
Conf.	**0.50**	0.47	**0.90**	0.89
Time	**47.4s**	47.8s	**38.1s**	38.6s

Table 1: User study results.

To explain this finding, it is worth noting that SymWriter pictures include function words, whose icons are abstract but distinct. We speculate that even though none of our subjects were trained to recognize these function-word icons, the native speakers are more accustomed to the English syntactic structure, so they may be able to transliterate those icons back to words. In an ABC layout, the sentence order is mostly removed, and some phrases might be omitted due to the O tag. Thus native speakers do not get as many syntactic hints. On the other hand, non-native speakers do not have the same degree of built-in English syntactic knowledge. As such, they do not gain much from seeing the whole sentence sequence including function-word icons. Instead, they may have benefited from the ABC layout's added organization and potential exclusion of irrelevant icons.

If this reasoning holds, it has interesting implications for viewers who have lower English literacy: they might take away more meaning from a semantically structured layout like ABC. Verifying this is a direction for future work.

Finally, it is interesting that all subjects feel more confident in their responses to ABC layouts than linear layouts, and, despite their added complexity, ABC layouts do not require more response time than linear layouts.

[5] Using a Mann-Whitney rank sum test, the difference in native speakers' METEOR scores is statistically significant ($p = 0.003$), though the other differences are not (native BLEU, $p = 0.085$; non-native METEOR, $p = 0.172$; non-native BLEU, $p = 0.170$). Nevertheless, we observe some evidence to support our hypothesis that non-native speakers benefit from the ABC layout, and we intend to conduct follow-up experiments to test the claim further.

6 Conclusions

We proposed a semantically enhanced picture layout for pictorial communication. We formulated our ABC layout prediction problem as sequence tagging, and trained CRF models with linguistic features including semantic role labels. A user study indicated that our ABC layout has the potential to facilitate picture comprehension for people with limited literacy. Future work includes incorporating ABC layouts into our pictorial communication system, improving other components, and verifying our findings with additional user studies.

Acknowledgments

This work is supported by NSF IIS-0711887, and by the Wisconsin Alumni Research Foundation.

References

Adorni, G., M. Di Manzo, and G. Ferrari. 1983. Natural language input for scene generation. In *ACL*.

Baker, C. F., C. J. Fillmore, and J. B. Lowe. 1998. The Berkeley FrameNet Project. In *COLING*.

Brants, T. and A. Franz. 2006. Web 1T 5-gram version 1.1. Linguistic Data Consortium, Philadelphia.

Brown, D. C. and B. Chandrasekaran. 1981. Design considerations for picture production in a natural language graphics system. *SIGGRAPH*, 15(2).

Chandrasekar, R., C. Doran, and B. Srinivas. 1996. Motivations and methods for text simplification. In *COLING*.

Ciaramita, M. and Y. Altun. 2006. Broad-coverage sense disambiguation and information extraction with a supersense sequence tagger. In *EMNLP*.

Clay, S. R. and J. Wilhelms. 1996. Put: Language-based interactive manipulation of objects. *IEEE Computer Graphics and Applications*, 16(2).

Coyne, B. and R. Sproat. 2001. WordsEye: An automatic text-to-scene conversion system. In *SIGGRAPH*.

Fleiss, J. L. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5).

Gildea, D. and D. Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3).

Hehner, B. 1980. *Blissymbols for use*. Blissymbolics Communication Institute.

Johansson, R., A. Berglund, M. Danielsson, and P. Nugues. 2005. Automatic text-to-scene conversion in the traffic accident domain. In *IJCAI*.

Joshi, D., J. Z. Wang, and J. Li. 2006. The story picturing engine—a system for automatic text illustration. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 2(1).

Klein, D. and C. D. Manning. 2003. Accurate unlexicalized parsing. In *ACL*.

Lafferty, J., A. McCallum, and F. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*.

Lavie, A. and A. Agarwal. 2007. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *Second Workshop on Statistical Machine Translation*, June.

McCallum, A. K. 2002. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu.

Mihalcea, R. and B. Leong. 2006. Toward communicating simple sentences using pictorial representations. In *Association of Machine Translation in the Americas*.

Palmer, M., D. Gildea, and P. Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1).

Papineni, K., S. Roukos, T. Ward, and W. Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL*.

Porter, M. F. 2001. Snowball: A language for stemming algorithms. http://snowball.tartarus.org/.

Pradhan, S., W. Ward, K. Hacioglu, J. Martin, and D. Jurafsky. 2004. Shallow semantic parsing using support vector machines. In *HLT/NAACL*.

Sampson, G. 2003. The structure of children's writing: Moving from spoken to adult written norms. In Granger, S. and S. Petch-Tyson, editors, *Extending the Scope of Corpus-Based Research*. Rodopi.

Vickrey, D. and D. Koller. 2008. Sentence simplification for semantic role labeling. In *ACL*. To appear.

Widgit Software. 2007. SymWriter. http://www.mayer-johnson.com.

Yamada, A., T. Yamamoto, H. Ikeda, T. Nishida, and S. Doshita. 1992. Reconstructing spatial image from natural language texts. In *COLING*.

Zhu, X., A. B. Goldberg, M. Eldawy, C. Dyer, and B. Strock. 2007. A Text-to-Picture synthesis system for augmenting communication. In *AAAI*.