

# Compositional Language Modeling for Icon-Based Augmentative and Alternative Communication

Shiran Dudy    Steven Bedrick

Center for Spoken Language Understanding  
Oregon Health & Science University  
3181 S.W. Sam Jackson Park Rd.  
Portland, Oregon, USA  
{dudy,bedricks}@ohsu.edu

## Abstract

Icon-based communication systems are widely used in the field of Augmentative and Alternative Communication. Typically, icon-based systems have lagged behind word- and character-based systems in terms of predictive typing functionality, due to the challenges inherent to training icon-based language models. We propose a method for synthesizing training data for use in icon-based language models, and explore two different modeling strategies.

## 1 Introduction

Individuals who experience speech and language impairments often are helped by Augmentative and Alternative Communication (AAC) techniques that facilitate the expression or comprehension of spoken or written language (Beukelman and Mirenda, 2005; American Speech Language Hearing Association et al., 2004; Fossett and Mirenda, 2007). Impairments may result from developmental disorders affecting speech and language (Cerebral Palsy, Down Syndrome, some forms of Autism Spectrum Disorder, etc.), or they may be caused by injury (stroke, traumatic brain injury, neurodegenerative diseases such as ALS, etc.). AAC interventions can take many forms, but a common goal is to provide users with a way to select symbols (words, phrases, etc.) for purposes of communication. The field of AAC groups interventions into "low-technology" (i.e., printed) or "high-technology" (i.e., computerized) devices; both are commonly used, and there are a number of factors that go in to the decision of which device to use (Iacono et al., 2011; Light and Drager, 2007). In some cases, devices produce speech (or written language) based on those selections, whereas in other cases, the goal of the device is

to support a user in producing their own speech.

In some cases, the unit of selection may be icon-based rather than word- or character-based. This is particularly common in devices used by children or individuals with impaired literacy, but is also common in adult use. Icon-based systems can have higher selection speeds, and can be easier for individuals with neuromuscular impairments to operate; there are a wide variety of symbol sets and symbol-based communication systems used (Patel, 2011).

Computerized text-based AAC devices often include some form of word prediction using a language model (see (Vertanen and Kristensson, 2011; Garay-Vitoria and Abascal, 2006) for overviews of this area). Icon-based systems often do not employ predictive features.<sup>1</sup> In part, this is because they typically rely on direct-selection or other input modalities; another barrier, is a lack of relevant linguistic training data. To our knowledge, there are no corpora of language produced using icon-based AAC systems.

In the present work, we apply modern language modeling techniques to a large-vocabulary icon set commonly used in AAC applications, but for which we have no in-domain (or even in-vocabulary) training data. We are building our model as support for a brain-computer interface (Orhan et al., 2012). In this input modality, the user has very limited control over item selection, making accurate language modeling critical.

We propose a method to generate language models and evaluate their performance by experimenting with a process to generate a trainable language modeling corpus. We also share an experi-

<sup>1</sup>One notable exception to this trend is the system used in SymbolPath (Wiegand and Patel, 2012b), which uses semantic frames to attempt non-sequential symbol prediction (Wiegand and Patel, 2012a). This work, however, was limited to a specific and small icon set.

mental setup for a new language modeling architecture. Our contributions in this paper are:

- A proposed approach to synthesize a pseudo-corpus with which to learn language models from a corpus-less symbol set
- An experimental evaluation of the impact of various pieces of our corpus synthesis methodology on icon prediction accuracy
- A preliminary attempt to apply a novel language model architecture suitable for icon-based, open-vocabulary AAC applications

## 2 Symbolstix dataset

The Symbolstix (Clark, 1997) icon set is used in several commercial AAC applications. Each icon includes an image, and is associated with metadata information that textually describes the meaning assigned to the icon at hand. An image in this set can represent a single word, a phrase, or a syntactic modifier (such as “plural”). The images cover 48 major topics such as actions, technology, nature etc. They vary in meaning, demonstrating abstract concepts and tangible ones; they also present different levels of complexity and details. The metadata primarily describe the associated term the image corresponds to, its synonyms, and its translation to different languages.

The Symbolstix icon set was designed for use by communities who are in need of icon-based communication, such as children with communication disabilities, TBI patients, etc. One commercial application of the current icon-set is a news-cast aimed at adult consumers; another is a communications platform for children with Cerebral Palsy. The set of 34k icons is in practice broad enough to reflect those needs. However, the creators of the system often do add new icons when requested by their user population.

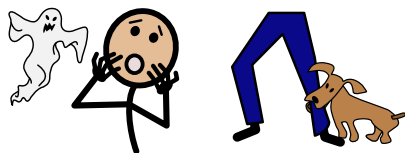


Figure 1: Example icons: *afraid* and *bite leg*

On the left side of Figure 1 is an example of a single-word icon representing the term *afraid*. This icon’s human-assigned topic category is “descriptives-feelings”. Note that many icons are mapped to multiple synonyms. Synonyms of

this icon are: *eerie*, *fear*, *feared*, *fearful*, *fears*, *frightened*, *Halloween*, *scary*, *terrified*, *upset*, and *nervous*. As such, an icon’s meaning is highly context-dependent.

The right-hand side icon in Figure 1 is an example of a two-word icon representing the concept *bite leg*. This icon’s topic is of “actions”. Synonyms of this icon are: “bad day” and “dog bite”. As observed in Figure 1, the nature of the Symbolstix leans toward conversational concepts of spoken language, due to their intended use in AAC.

Symbolstix contains 34,837 icons. 13,951 of these icons are of a single word; several of these are duplicates<sup>2</sup>, only 12,434 of the single word icons are unique terms. In our experiment, to avoid redundancy, we used this set of unique terms. We chose the unique term-icon pair (from its non-unique group) that had the richest metadata and the highest overlap (within its group) to represent a concept. This step was essential to reduce complexity; however, it did introduce a limitation to our approach, which we discuss in section 5.

Notably, the Symbolstix corpus comes with no dataset that demonstrates the intended usage of icons to construct a proper sentence. Ideally, we would use a dataset of icon sentences for language modeling, as it would enable learning icon sequences and by that to infer the language rules from its patterns. In the next section, we describe how we were able to overcome that obstacle.

## 3 Experimental Setup

As mentioned in Section 2, the Symbolstix data set of icons has no sentence-like corpus from which icon sequences can be readily learned to form a language model of icons. We attempted to synthesize an icon corpus by beginning with a textual corpus, and “projecting” our icons into the text space using pre-trained word embeddings using the methodology described below. Since each icon is accompanied by metadata containing human-edited synonym lists, it was natural to represent icons as some composition of their synonyms in a vector space.

In this manner, we embedded our icons in text, and created pseudo-sequences (“icon sentences”). This solution is not without problems, first among them the issue is that our icon sentences may not

<sup>2</sup>A single lexical item may be represented several times; for example, there are several variants of “glue”, with different shapes of bottles and patterns of labels.

represent realistic examples of how the Symbolstix icons are meant to be used. Rather, they might instead represent the icons as subjected to the language conventions found in oral or written language. For example comparing a possible icon sequence to the English language might look like:

Icon:  $\langle I \rangle \langle go \rangle \langle here \rangle \langle past \rangle$

English:  $\langle I \rangle \langle went \rangle \langle here \rangle$ , or

English:  $\langle the \rangle \langle dogs \rangle \langle are \rangle \langle at \rangle \langle home \rangle$

Icon:  $\langle the \rangle \langle dog \rangle \langle plural \rangle [\langle be \rangle \text{ optional}] \langle at \rangle \langle home \rangle$

Some of the terms may disappear in translation, while others are added.

Another question is whether it is possible to fully represent an icon as the sum of its synonyms; or, put another way, whether the ways in which an icon’s synonyms are used in written language can capture the totality of an icon’s meaning. For example, the icon in figure 1 does not precisely mean “afraid”, but rather refers to a more general concept. This is why we chose to explore a compositional approach to representing icon meaning in a continuous space. Finally, how should we handle sentences in our textual data set that are not fully representable using icons from Symbolstix?

## Data Preprocessing

*Icon Representation:* In order to construct an icon language model, we needed to find a way to represent our icon vocabulary in a continuous embedding space (following the lead of [Kiros et al. \(2015\)](#)). Lacking a corpus that included icons, we were unable to directly train “icon embeddings” from data. Instead, we attempted to “project” our icons into a word embedding space.

We experimented with two different approaches of word embedding (see section 4), and, for each icon, generated icon embeddings by averaging<sup>3</sup> the word embeddings of the icon’s synonyms (as specified by the Symbolstix metadata). Note that a different choice of icon set would have resulted in a different embedding space and language model. However, this basic approach describes a generic process to produce models from a corpus-less symbol-set, and should translate to other situations.

*Textual Datasets:* We next took a textual corpus (see section 4), and identified terms that could be replaced with icon embeddings. In this work, we relied on a relatively simple strategy of mechanically substituting icons for words based on

<sup>3</sup>We also experimented with summation, and observed no meaningful difference.

their Symbolstix metadata. In other words, instances of the word “wetland” in the corpus would be replaced by the icon with “wetland” in its list of synonyms or descriptor terms. Note that this icon’s embedding would contain information about other words associated with that icon (e.g. “swamp”). We discuss some practical considerations around polysemous words and icons in section 4. This step forms a dataset of embedding sequences, which we then used as a corpus for learning a language model. The resulting corpus was then fully representable by embeddings (both of the words and icons).

## Training

For our language model, we used a standard RNN architecture with two hidden LSTM layers, a linear, and a final softmax layer that predicts the term’s index, trained with cross entropy loss function (seen in Equation 1). The model’s input is of the generated embeddings and as such contained no explicit embedding layer.

$$L_i = \sum_{j \in J} T(i, j) \log(P(i, j)). \quad (1)$$

The icons in this project are aimed at patients who would use them as a means to communicate. It is very likely that the icon language that would be formed by these patients would share similar characteristics with spontaneous speech or informal language since this is the type of communication we have with a caregiver, family member, or a friend. Knowing this, we used the SubtlexUS ([Brysbaert and New, 2009](#)) corpus (made up of subtitles from movies and television) as a proxy for a corpus of spontaneous speech that contained 6, 043, 188 sentences.<sup>4</sup>

## Model Evaluation

We evaluated our language model using three different metrics:

- Mean Reciprocal Rank (MRR) of the “correct” predicted icon as seen in Equation 2

$$MRR = \frac{1}{|Q|} \sum_{i \in Q} \frac{1}{rank_i}, \quad (2)$$

$Q$  represents the token events of the target. The choice of MRR metric was to internally

<sup>4</sup>See ([Vertanen and Kristensson, 2011](#)) for an extensive discussion on issues surrounding corpus selection for AAC applications.

look at the rank of the target, rather than to binary classify it for whether it was accurately predicted in the first rank.

- **Accuracy@ $k$ :** The percentage of predictions in which the “correct” icon was within the top  $k$  predictions. The choice in ACC@ $k$  was to inform about the quality of the prediction generated by the models. We have chosen ACC@1 to crudely understand whether the first choice was correct, and ACC@10 to get a sense of the prediction quality given that a user may be able to choose from a limited list (depending on the user interface), and also to model the notion that different users may choose different words in a similar context (and so there is not a single correct word in reality).

## 4 Experiments

We performed three sets of experiments. The first explored the effect of different approaches to word embedding, the second explored the effects of either including or excluding non-icon terms in model training, and the third looked at the effects of other (non-Subtlex) text corpora. For both approaches to word embedding, we used pre-trained word vectors. The pretrained set is the source for generating the icon embeddings.

Both the icon and the pretrained embeddings replace the terms in the textual data with their corresponding vectors to generate an embedding corpus. All our experiments contained the same number of pretrained vectors as well as icon vectors. If both vector sets contained the same term, the icon embedding was used. The textual dataset was tokenized and punctuation was removed. Each of these experiments was held in a 5 fold cross validation fashion. The process to generate the corpus from which language models are learned is described in Figure 2.

This process shows also the three different modules we experimented with: the pretrained corpus, which forms the icon embeddings; the icon set that forms (with or without the pretrained set (therefore the ‘switch’ between pretrained embeddings to textual dataset)) the textual embedding; and the textual dataset that provides the sequences of symbols to generate the textual embedding.

### Experiment 1: Pretrained Vectors

The Pretrained embeddings in our experiment are used both to construct the icon representations and

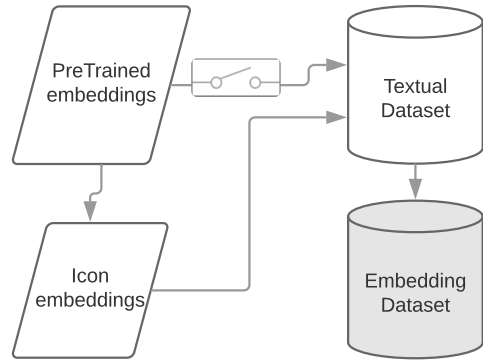


Figure 2: Block diagram to generate training set for language modeling

to represent the textual corpus. In our setting we have explored the Global Vectors for Word Representation (Glove) (Pennington et al., 2014) set consisting of 400,000 uncased entries and trained on Wikipedia 2014 and Gigaword version 5 ( $\approx 6B$  tokens) with 50 embedding dimensions, and compared it to Context2Vec (c2v) (Melamud et al., 2016) trained on ukWaC corpus and MSCC ( $\approx 2B + \approx 50M$  tokens) consisting of 160,563 uncased entries, with 600 embedding dimensions. To maintain a controlled environment the textual dataset remained fixed, the vocabulary of pre-trained token types was identical ( $n=69,840$ ) as well as the icon list ( $n=6,934$  term types) irrespective of the dataset used. A term in the dataset was prioritized to be replaced with an icon term first, then if not found, with an icon synonym, with a pretrained representation, and finally, provided no alternative, a term was replaced with an  $\langle unk \rangle$  vector representation.

metric	c2v	Glove
MRR	0.85 (0.00)	0.85 (0.00)
ACC@1 (%)	50.99 (0.03)	49.29 (0.04)
ACC@10 (%)	90.51 (0.01)	90.29 (0.01)

Table 1: Effect of word embedding method

Table 1 contains the experimental results of evaluations run on models trained with Glove and c2v vectors, averaged across five folds of cross-validation on the SubtlexUS dataset.

After controlling for the number of icon- and pretrained-term types as well as for the textual corpus, Table 1 shows that there are no meaningful differences resulted from the pretrained vectors type. The dataset the vectors were trained on as

well as the method by which the vectors were generated had no observable impact on the language model performance.

While Experiment 1 covers one aspect of comparing differences between different word embeddings, when choosing a pretrained set of vectors to use and generate icons from, there may be additional considerations. The coverage of the pretrained set is essential to produce icon representations, but also is important for terms in the textual dataset that cannot be represented with icons (which are then replaced with a pretrained vector if found) as described in Experiment 1. The pretrained set coverage with regards to the icon set is measured not only by the total number of icon representations that were generated from the pretrained set, but also by how well each icon captures the broad meaning it stands for. Since each icon is likely to have its name and synonyms composed together to represent it (as described in 3 in Data Preprocessing part), an optimal pretrained set would contain representations for all these terms. As for the textual dataset, an optimal coverage of the text with the pretrained list ideally would consist of a large number of term types, but also term events that appear in the dataset.

## Experiment 2: Icon Symbols Constraint

Ideally, the corpus to learn language models from would consist of the icon vocabulary solely since the goal is to construct an Icon language model. We therefore, experimented with transforming our synthetic corpus to *only* include terms representable using icon vectors (“pure”) and compared LM and prediction accuracy with the original, “non-pure” results. We used Glove and c2v vectors in our experiment presented in Table 2.

Table 2 describe an averaged five fold cross validation experiment of SubtlexUS with icon only embeddings referred by the “pure” experiment.

metric	c2v	Glove
MRR	0.33 (0.00)	0.33 (0.00)
ACC@1 (%)	46.79 (0.06)	45.72 (0.06)
ACC@10 (%)	54.92 (0.01)	54.29 (0.04)

Table 2: Effect of Icon embedding representation on SubtlexUS mean(standard deviation)

Table 2 describes a similar pattern to Table 1 as there was no meaningful change in the final icon language models’ performances due to the “pure”

condition. We do note that there was a slight advantage to c2v embeddings which seemed to be predicting more correctly the target.

The “pure” condition resulted in a relatively smaller prediction accuracy. On the one hand, this may be surprising evidence, as it is reasonable to think that a smaller and more focused vocabulary set would result in an improved language model performance. We assume that the reduction in vocabulary size caused as a result of employing icons solely created short, sparse, and uncommon patterns of sequences, which limited the models’ ability to learn and predict accurately.

Under the “pure” condition, the model vocabulary consists solely of the icon set itself, whereas in the “non-pure” condition, the model vocabulary consists of the icon set as well as the pretrained embeddings together with *<unk>* terms. While we can not directly compare the two experiments (1 and 2) we can share our considerations when choosing to generate language models purely based on icons.

To support our explanation for Table 2 interpretation, we conducted a qualitative test and looked into to the actual sentences produced by the icons in isolation, asking whether these sentences created “meaningful” (or at least useful) messages for LM training.. This might be helpful to get a deeper perspective on the corpus created and assist in making design choices. Here is an example:

“non-pure”: *<your> <warning> <did> <not> <work>*

“pure”: *<your> <warning> <not> <work>*

Arguably, the main message was conveyed in this sentence, while in the following:

“non-pure”: *<they> <did> <n’t> <use> <mud> <they> <used> <sod>*

“pure”: *<they> <use> <they>*

the essence is gone. While it is not feasible to qualitatively look at every sentence, one may consider comparing the amount of tokens prior to elimination and post, under the assumption that the greater the loss, the more likely that the quality of solely using the icon-set becomes a concern.

We would like to note that in Experiment 2 in particular, we used the same simulated “icon language” for both training and evaluation. An ideal evaluation of our approach to producing synthetic in-domain training data would have been evaluating the language models trained on simulated icon language on “real” text composed using icons. As

we did not have such a useful resource, it is important to observe this as a limitation of the current experiment.

### Experiment 3: Textual Corpus

In our system, the role of the textual corpus is to provide the language model with training data regarding patterns of word (“icon”) use. Ideally, we would use a corpus made of symbols that represent the type of content and structure an AAC user would produce. Finding an AAC-oriented corpus that would be big enough to train was a hurdle, and so for our previously-described experiments, we relied on SubtlexUS. While not ideal, this corpus was closer to spontaneous speech than, say, a newswire corpus would have been, and featured smaller and more manageable sentences that we hoped would withstand being converted to pseudo-icon representations.

That said, we did wish to investigate the utility of using an existing corpus that was designed to be closer to AAC-style speech. [Vertanen and Kristensson \(2011\)](#) produced such a corpus, consisting of 6,142 sentences produced by Amazon Mechanical Turk users who were paid to generate plausible sentences and to evaluate the plausibility of other sentences generated by other workers.

This corpus, while valuable, was too small for use with our language modeling approach. Following [Vertanen and Kristensson’s](#) insight that “short text” such as that seen in online media such as Twitter, etc. might be a good proxy for true AAC-style speech, we therefore mixed the AAC-style corpus with second corpus, this one consisting of modified SMS messages. The second corpus was from [Chen and Kan \(2013\)](#) and consisted of 18,042 SMS messages, and was originally constructed for experiments in text normalization. As such, it includes messages written in heavily-abbreviated forms as well as “cleaned up” versions of each message, written in something approximating “standard” English orthography. We used this subset of the corpus in the hopes that its short, informal, and speech-like sentences would complement the AAC-style corpus. Our goal was to assemble a corpus containing language that is as close as possible to what would be produced by actual users of an AAC system. We then repeated the language modeling experiments conducted earlier on this hybrid corpus, using identical procedures and evaluation metrics.

Table 3 and 4 tell a similar story to one an-

metric	c2v	Glove
MRR	0.38 (0.00)	0.33 (0.01)
ACC@1 (%)	56.41 (0.52)	47.13 (2.81)
ACC@10 (%)	60.44 (0.42)	54.51 (2.37)

Table 3: Textual corpus (AAC-SMS) mean(standard deviation)

metric	c2v	Glove
MRR	0.37 (0.01)	0.34 (0.00)
ACC@1 (%)	53.25 (2.11)	47.73 (1.46)
ACC@10 (%)	59.06 (1.29)	56.00 (1.40)

Table 4: Textual corpus (AAC-SMS/pure) mean(standard deviation)

other: the models trained using c2v embeddings outperformed the models that used Glove embeddings, which is different from what we observed in the previous experiments— though with this corpus, the overall performance numbers were much lower than with the original, larger corpora. The reason for overall low performance was probably due to the very small size of the AAC-SMS corpus, and possible overfitting as a result. Digging more deeply into our data, we examined the individual cross-validation results at the fold level, thinking that perhaps the results were unstable due to the relatively small data set which indeed seem to be the case.

Nevertheless, in an attempt to indirectly evaluate different models, while the AAC experiments had substantially higher variance across folds than did the SubtlexUS experiments, the differences between the two approaches do appear to be real. Ultimately, we note that the substantial difference in corpus size between SubtlexUS and our AAC-SMS corpus make it difficult to draw any firm conclusions, and investigating this issue further will be a component of our future work in this area.

## 5 Conclusions

This work is a first step towards the development of language models for an icon set that has no corresponding corpus, but there remains much to be done. One limitation of this work is that, even after being projected into an icon space, our synthetic training data is somewhat different from actual icon-based language produced by AAC users. That said, we did try to overcome this limitation by experimenting with a corpus designed to be much

closer to actual AAC-style language, though at a substantial cost in terms of corpus size. This is a common problem in AAC research in general, and our immediate next steps will focus on developing more naturalistic training corpora (following the lead of Vertanen and Kristensson (2011), who faced similar challenges).

Another important limitation is that we have not solved the problem of icons that represent multi-word expressions or phrases. This will be a major area our future work, for two reasons. First, many important icons fall into this category. Second, one of the advantages of icon-based AAC is increased speed of communication, and collapsing multi-word expressions to a single icon would enable substantial improvements.

A second area of future work will look at ways to capture and express morphology. Our icon set includes icons that can be used to indicate tense, plurality, etc., but our current approach to corpus processing and term substitution/composition does not take advantage of such information. We intend to explore ways to directly represent morphological/inflectional information in the input side of our models, and in doing so make better use of our icon system.

A final limitation of this work is that our approach to selecting icons had the unfortunate side effect of ignoring polysemy: the set of icons that we worked with here was restricted to a single sense of polysemous words. This means that some possibly-useful icons were excluded, which could have consequences for anybody actually using our system for communication. Consider the word “cheer”, which can be either a verb or a noun, and in both cases has multiple meanings. There are several icons in Symbolstix that capture different usages of the word, but our current approach only uses one. This will be another active area of future work, and we expect our solution to this problem to tie in with our solution to the issue of multi-word expressions.

Our evaluations thus far have been system-oriented, and have tried to measure the model’s performance. Our MRR and accuracy results have provided us with an internal view for where our models were performing as desired as well as identified areas where they fall short. The next step will be to integrate our language model with the rest of our AAC platform and begin working with real end-users. We anticipate that this will

guide much of our future work on this problem.

## Acknowledgments

We thank the anonymous DeepLo workshop reviewers for their valuable feedback and comments. We also thank our clinical collaborators in OHSU’s Institute on Development & Disability: Betts Peters, Brandon Eddy, and Dr. Melanie Fried-Oken, as well as our collaborators at Northeastern University, in the laboratories of Drs. David Smith and Deniz Erdogmus. Research reported in this paper was supported by the National Institute on Deafness and Other Communication Disorders of the NIH under awards R01DC009834 and R56DC015999.

## References

- American Speech Language Hearing Association et al. 2004. Roles and Responsibilities of Speech-Language Pathologists with Respect to Augmentative and Alternative Communication: Technical Report .
- David R. Beukelman and Pat Mirenda. 2005. *Augmentative & Alternative Communication: Supporting Children & Adults with Complex Communication Needs*. Paul H. Brookes Publishing Co., 3rd edition.
- Marc Brysbaert and Boris New. 2009. Moving Beyond Kučera and Francis: A Critical Evaluation of Current Word Frequency Norms and the Introduction of a New and Improved Word Frequency Measure for American English. *Behavior Research Methods* 41(4):977–990.
- Tao Chen and Min-Yen Kan. 2013. Creating a live, public short message service corpus: the nus sms corpus. *Language Resources and Evaluation* 47(2):299–335.
- Jacque Clark. 1997. *Symbolstix*. News 2 You. <https://www.n2y.com/symbolstix-prime>.
- Brenda Fossett and Pat Mirenda. 2007. Augmentative and Alternative Communication. *Handbook of Developmental Disabilities* pages 330–348.
- Nestor Garay-Vitoria and Julio Abascal. 2006. Text prediction systems: a survey. *Universal Access in the Information Society* 4(3):188–203.
- Teresa Iacono, Katie Lyon, and Denise West. 2011. Non-Electronic Communication Aids for People with Complex Communication Needs. *International Journal of Speech-Language Pathology* 13(5):399–410.

- Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-Thought Vectors. In *Advances in Neural Information Processing Systems*. pages 3294–3302.
- Janice Light and Kathryn Drager. 2007. AAC Technologies for Young Children with Complex Communication Needs: State of the Science and Future Research Directions. *Augmentative and Alternative Communication* 23(3):204–216.
- Oren Melamud, Jacob Goldberger, and Ido Dagan. 2016. context2vec: Learning Generic Context Embedding with Bidirectional LSTM. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*. pages 51–61.
- U. Orhan, K. E. Hild, D. Erdogmus, B. Roark, B. Oken, and M. Fried-Oken. 2012. Rsvp Keyboard: An EEG Based Typing Interface. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. pages 645–648.
- R. Patel. 2011. Message Formulation, Organization, and Navigation Schemes for Icon-Based Communication Aids. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. pages 5364–5367.
- Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. pages 1532–1543.
- Keith Vertanen and Per Ola Kristensson. 2011. The imagination of crowds: Conversational aac language modeling using crowdsourcing and large data sources. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Stroudsburg, PA, USA, EMNLP '11, pages 700–711. <http://dl.acm.org/citation.cfm?id=2145432.2145514>.
- Karl Wiegand and Rupal Patel. 2012a. Non-syntactic word prediction for aac. In *Proceedings of the Third Workshop on Speech and Language Processing for Assistive Technologies*. Association for Computational Linguistics, Stroudsburg, PA, USA, SLPAT '12, pages 28–36. <http://dl.acm.org/citation.cfm?id=2392855.2392860>.
- Karl Wiegand and Rupal Patel. 2012b. Symbol-path: A continuous motion overlay module for icon-based assistive communication. In *Proceedings of the 14th International ACM SIGACCESS Conference on Computers and Accessibility*. ACM, New York, NY, USA, ASSETS '12, pages 209–210. <https://doi.org/10.1145/2384916.2384957>.