

DEEP MULTIMODAL SEMANTIC EMBEDDINGS FOR SPEECH AND IMAGES

David Harwath and James Glass

MIT Computer Science and Artificial Intelligence Laboratory
Cambridge, Massachusetts, 02139, U.S.A
{dharwath, glass}@mit.edu

ABSTRACT

In this paper, we present a model which takes as input a corpus of images with relevant spoken captions and finds a correspondence between the two modalities. We employ a pair of convolutional neural networks to model visual objects and speech signals at the word level, and tie the networks together with an embedding and alignment model which learns a joint semantic space over both modalities. We evaluate our model using image search and annotation tasks on the Flickr8k dataset, which we augmented by collecting a corpus of 40,000 spoken captions using Amazon Mechanical Turk.

Index Terms— Neural networks, multimodal semantic embeddings

1. INTRODUCTION AND RELATED WORK

Conventional automatic speech recognition (ASR) systems utilize training data in the form of speech audio with parallel text transcriptions. In this paper, we investigate what is possible to do if those text transcripts were replaced with relevant visual images. Given a dataset comprised of image scenes with accompanying spoken audio captions segmented at the word level, we propose a model capable of learning to associate spoken instances of the word "dog" with images of dogs, to name just one example. Our model relies on a pair of convolutional neural networks (CNNs), one for images and another for speech, along with an alignment and embedding model. The outputs of the networks provide fixed-dimensional representations of variable-sized visual objects and spoken words, which are then mapped into a shared semantic embedding space. This allows us to align the words in the captions to the objects they refer to in the image scene. While a large body of research on jointly modeling images and text exists in the literature, we are not aware of any prior work that models the semantics of images and speech directly on the audio signal level.

Multimodal modeling of images and text has been an extremely popular pursuit in the machine learning field during the past decade, with many approaches focusing on accurately annotating objects and regions within images. For example, Barnard et al. [1] relied on pre-segmented and labelled images to estimate joint distributions over words and objects,

while Socher [2] learned a latent meaning space covering images and words learned on non-parallel data. While these approaches focused on improving the identification of visual objects from a pool of predefined classes, other research has studied the problem of aligning text to the images or videos they describe. For example, Kong et al. [3] took visual scenes with high level captions, parsed the text, detected visual objects, and then aligned the two modalities with a Markov random field. Lin et al [4] aligned semantic graphs over text queries to relational graphs over objects in videos to perform natural language video search. Matuszek et al. [5] employed separate classifiers over text and visual objects that shared the same label sets.

A related problem is that of natural language caption generation. While a large number of papers have been published on this subject, recent efforts using recurrent deep neural networks [6, 7] have made tremendous progress and generated much interest in the field. While our work in this paper does not aim to generate captions for images, it was originally inspired by the text-to-image alignment models presented by Karpathy in [6, 8]. In [6], Karpathy uses a refined version of the alignment model presented in [8] to produce training exemplars for a caption-generating RNN language model that can be conditioned on visual features. Through the alignment process, a semantic embedding space containing both images and words is learned. Other works have also attempted to learn multimodal semantic embedding spaces, such as Frome et al. [9] who trained separate deep neural networks for language modeling as well as visual object classification. They then embedded the object classes into a dense word vector space with the neural network language model, and fine-tuned the visual object network to predict the embedding vectors of the words corresponding to the object classes. This paper shares much of the same spirit as prior work on semantic embedding of images and words, but with a key difference - instead of dealing with text at the orthographic level, we learn a model which can derive meaning directly from spoken audio.

2. MODEL DESCRIPTION

Our overarching goal is to be able to represent examples of spoken words, alongside examples of visual objects, as points

in a single, high dimensional vector space. For example, in this vector space, we want different spoken examples of the word “dog” to neighbor one another, and also to neighbor image crops containing dogs. In order to do this, we require some means to transform variable sized image crops as well as variable duration audio waveforms into fixed dimensional vector representations. Further, we also require some way of coaxing these vectors into taking on the the property that semantically similar images and words neighbor one another. To achieve this, we employ two separate neural network architectures, one for images and one for audio, which we then marry together with an embedding alignment model.

2.1. Region Convolutional Neural Network

In order to detect a set of candidate regions in an image which are likely to contain meaningful objects, we use the Region Convolutional Neural Network (RCNN) model [10]. The RCNN object detector works by first using selective search [11] to build a large list of proposal regions, typically numbering in the thousands for a given image. Each proposal region is then fed into a CNN object classifier, which is used to extract the activations of the penultimate layer of neurons in the network. These activations form a fixed-dimensional (4096 in [10], as well as our work) feature vector representation of each proposal region. A set of one-versus-all support vector machines are then used to calculate detection scores over some set of classes for each region, and highly overlapping regions with similar classification scores are merged. Finally, the remaining set of regions can be ranked in order of their maximum classification score across all classes. In our work, we follow [6] and take the top 19 detected regions along with the entire image frame, resulting in 20 regions per image. We use the $d_I = 4096$ dimensional RCNN feature vectors to represent each region, which we will refer to as $\mathcal{V} = \{v_i | i = 1 \dots 20\}$

2.2. Spectrogram Convolutional Neural Network

Previous efforts [6, 7] to perform semantic alignment of text to objects in image scenes have benefited from the fact that text is naturally segmented into words, and all instances of the same word share the same orthography. On the other hand, segmenting continuous speech into words is nontrivial, and different spoken instances of the same underlying word will inevitably differ in not only their duration, but also in their acoustic feature representations as influenced by factors such as the microphone and speaker characteristics and the context in which the word was spoken.

While a speech recognition system is a reasonable solution for building a spoken interface for natural language image retrieval systems such as the one described in [6], in this work we are more interested in investigating the potential of neural networks to learn meaningful semantic representations which operate directly on the feature level. However, tasking

our system with also performing word segmentation on the audio stream significantly complicates the problem at hand. We choose to take a step back from the text-based framework by pre-segmenting each spoken caption into a sequence of audio waveforms, each containing a single ground-truth word, and then throwing away the word identity of each segment.

In [12], the authors trained a CNN isolated word recognizer and utilized it for N-best recognition hypothesis re-ranking; here, we propose to use a similar CNN to model the spectrogram of each isolated word in the image captions. Standard CNNs expect their inputs to be of a fixed size, so in order to accommodate our variable duration words we follow [12] and choose to embed their spectrograms in a fixed duration window, applying zero-padding and truncation when necessary. While [12] found that a 2 second window was sufficient to capture the duration of 97% of the words in their corpus, in our case a 1 second long window is long enough to capture 99.9% of the words appearing in our data.

To create the spectrogram representing each word, we begin by performing forced-alignment of the audio to its ground truth text transcription in order to determine word boundary information. Next, we apply a standard 25 millisecond window with a 10 millisecond shift to each word utterance, extracting log energy filterbank features for each window using 40 filterbanks spaced along the mel scale. Next, we subtract the mean value and then apply variance normalization to the entire spectrogram. Finally, we either pad with zeros or truncate equally on both sides to force the spectrogram to have a width of 100 frames, or 1 second. Figure 1 shows an example of what the input data to the network looks like for an instance of the word “strategists”.

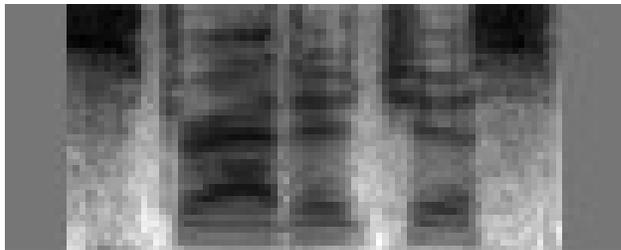


Fig. 1. Log mel filterbank spectrogram of the word “strategists”

We rely on the Caffe [13] toolkit to train our networks and extract the word spectrogram features. Our CNN architecture is as follows:

1. Pixel-by-pixel mean image spectrogram subtraction, with the mean spectrogram estimated over the entire training set;
2. Convolutional layer with filters sized 5 frames by 40 features with a stride of 1, vertical padding of 1 pixel on both the top and bottom, and 64 output channels with a ReLU nonlinearity;

3. Local response normalization of width 5, $\alpha = 0.0001$, and $\beta = 0.75$;
4. Max pooling layer of height 3, width 4, vertical stride 1, and horizontal stride 2;
5. Two fully connected layers of 1024 units each, with a dropout ratio of 0.5 and ReLU nonlinearities;
6. A softmax classification layer

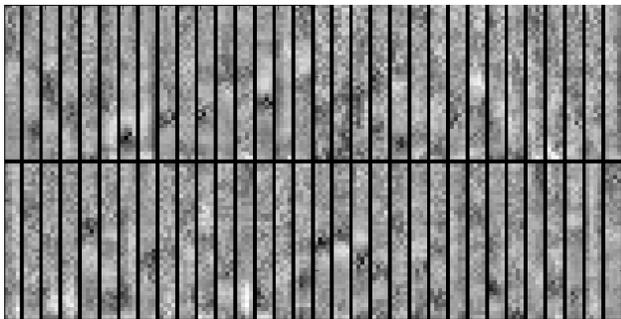


Fig. 2. 64 learned filters for the spectrogram CNN

To extract vector representations for each word in some image caption, we feed the word’s spectrogram through the network and discard the softmax outputs, retaining only the activations of the $d_W = 1024$ dimensional fully connected layer immediately before the classification layer. For a given caption, we will refer to these vectors as $\mathcal{W} = \{w_j | j \dots N_w\}$, where N_w is the number of words appearing in the caption.

2.3. Embedding Alignment Model

Given an image-caption pair and their corresponding object detection boxes and word spectrograms, our task is to align each word with one of the detection boxes found in the image. To do this, we adopt the transform model from [8] but with the objective function presented by [6]. However, we replace the text modelling side of Karpathy’s models with our word spectrogram CNN, enabling us to align the image fragments directly to segments of speech audio. We provide a brief overview of the alignment model and objective here.

Let $\mathcal{V} = \{v_i | i = 1 \dots 20\}$ be the set of d_I -dimensional vectors representing the activations of the penultimate layer of the RCNN for each detected image region, as described in Section 2.1. Also let $\mathcal{W} = \{w_j | j \dots N_w\}$ be the d_W -dimensional vectors representing the similar activations of the spectrogram CNN on each of the N_w words in the spoken caption. The job of the alignment model is to map all of the $v \in \mathcal{V}$ and $w \in \mathcal{W}$ vectors into a shared, h -dimensional space where semantically related words and images have a high similarity.

The alignment model is two-faceted, with separate transforms applied to the image vectors as well as the word spectrogram vectors. We use an affine transform, $y = W_m v + b_m$

to map an image vector v into the h -dimensional semantic embedding space. To map a word spectrogram vector w into that same embedding space, we use a nonlinear transform, $x = f(W_d w + b_d)$ where $f(z)$ is some element-wise nonlinear function. For the experiments in this paper, we set $f(z) = \max(0, z)$.

Motivated by the assumption that the spoken caption l for a given image k should contain words which directly reference objects in the image, Karpathy’s objective function tries to assign a high similarity to matching image-caption pairs by “grounding” each word vector to one or more image fragment vectors. The inner product similarity between a given word embedding and an image fragment embedding is used to measure the degree of grounding, and each word in caption l is given a score according to its maximum similarity across all image fragments from image k . An overall image-caption similarity score is then computed by summing the scores of all words in the caption, thresholded below at 0:

$$S_{kl} = \sum_{t \in g_l} \max_{i \in g_k} (0, y_i^T x_t), \quad (1)$$

where g_l denotes the set of image fragments in image l , and g_k is the set of word spectrograms in caption k .

In [6], Karpathy uses a max margin objective function which forces matching image-caption pairs to have a higher similarity score than mismatched pairs, by a margin. Given that S_{kk} denotes the similarity between a matching image-sentence pair, the cost is defined as:

$$\mathcal{C}(\theta) = \sum_k \left[\sum_l \max(0, S_{kl} - S_{kk} + 1) + \sum_k \max(0, S_{lk} - S_{kk} + 1) \right]. \quad (2)$$

In practice, we use stochastic gradient descent to optimize this cost function in terms of the parameters $\theta = \{W_m, b_m, W_d, b_d\}$.

3. DATA

Recent works on natural language image caption generation [6, 7] have utilized a number of datasets which contain images alongside human-generated text captions, such as Pascal, Flickr8k [14], Flickr30k [15], and MSCOCO [16]. However, none of these datasets include any speech data, so we decided to collect our own spoken audio for our experiments. Because of its manageable size and ubiquitousness in the previous literature, we choose to use the Flickr8k as the starting point for our data collection.

Flickr8k contains approximately 8,000 images captured from the Flickr photo sharing website, each of which depicts actions involving people or animals. Each image was annotated with a text caption by five different people, resulting in a total of 40,000 captions. To collect these captions the authors turned to Amazon’s Mechanical Turk, an online service

which allows requesters to post "Human Intelligence Tasks" (HITs). These HITs are then made available to anonymous, non-expert workers, or "Turkers", who can choose to complete the tasks for a small amount of money. We turned to Mechanical Turk to collect spoken audio recordings for each of the 40,000 captions from the Flickr8k dataset. We use the Spoke JavaScript framework [17] as the basis of our audio collection HIT. Spoke is a flexible framework for creating speech-enabled websites, acting as a wrapper around the HTML5 getUserMedia API while also supporting streaming audio from the client to a backend server via the Socket.io library. The Spoke client-side framework also includes an interface to Google's SpeechRecognition service, which can be used to provide near-instantaneous feedback to the Turker.

Figure 3 displays a screenshot of the audio collection interface we used in our HITs. A set of 10 random captions are displayed to the user, who can click the start/stop button to record their speech while they read each caption out loud. A playback button allows the Turker to listen to their own recordings and diagnose any problems with their microphone or environment. Spoke pipes the audio to the Google recognizer, checks the recognition result against the prompt, and notifies the user if their speech could not be recognized accurately. The Turker is then given the option to re-record the errorful caption. The HIT cannot be submitted until all 10 captions have been successfully recorded. In our experiments, we use a very simple metric for verification - 60% or more of the caption words must appear in the recognition result, regardless of ordering. We found this to be both lenient and sufficient - users rarely complained about the system correctly recognizing their speech, and 95.7% of the collected utterances were easily aligned to their caption text using our Kaldi [18] forced alignment system. The majority of the utterances flagged as unalignable were either empty or cut short, which we believe may have been due to client-server connection issues; the problematic utterances were recollected by another round of HITs. We paid the turkers 0.5 cents per spoken caption, resulting in a total cost of just over \$200 including Amazon's service fee. We collected speech from 183 unique Turkers, with the average worker completing 218 captions. There were a handful of Turkers who completed far more than the average number of captions, with the highest number collected from a single worker being 2,978.

To further verify the integrity of our collected audio data, we split the 40,000 utterances into a 30,000 utterance training set, a 5,000 utterance development set, and a 5,000 utterance testing set, covering a 8,918 word vocabulary. Our splits correspond with the training, validation, and testing splits given by [14]. We then used Kaldi to build a large vocabulary speech recognition system, adapting the standard Wall Street Journal recipe for a GMM/HMM + LDA + MLLT + SAT system for our data. We employed the training set to train the acoustic and language models, the CMU pronunciation lexicon, and the development set to tune the acoustic and lan-

guage model weights. The final word error rate of our system on the test set was 11.67%, providing another indication that our data is relatively high quality. In order to preprocess the Flickr8k data for our CNN, we employ this recognizer to force align the audio to the ground truth text transcripts and segment the audio at the word level.

Because the Flickr8k corpus contains a small number of images and captions relative to datasets such as ImageNet [19], we follow the example of [6] and use the off-the-shelf RCNN provided by [10] trained on ImageNet to extract the 4096-dimensional visual object embeddings. Similarly, we employ supervised pretraining for the word spectrogram CNN using the Wall Street Journal SI-284 split [20]. This set contains roughly 82 hours of speech, from which we extracted all instances of words occurring at least 10 times in the data. This gave us a total of 612,108 words covering a vocabulary of size 6,010, which we split 80/20 into training and testing sets. We used this data to train our word spectrogram CNN using the 6,010 word vocabulary as our output targets. Even though this training is supervised, 6,749 of the unique words appearing in the Flickr8k transcriptions (75% of the vocabulary) do not appear in the training set for the spectrogram CNN.

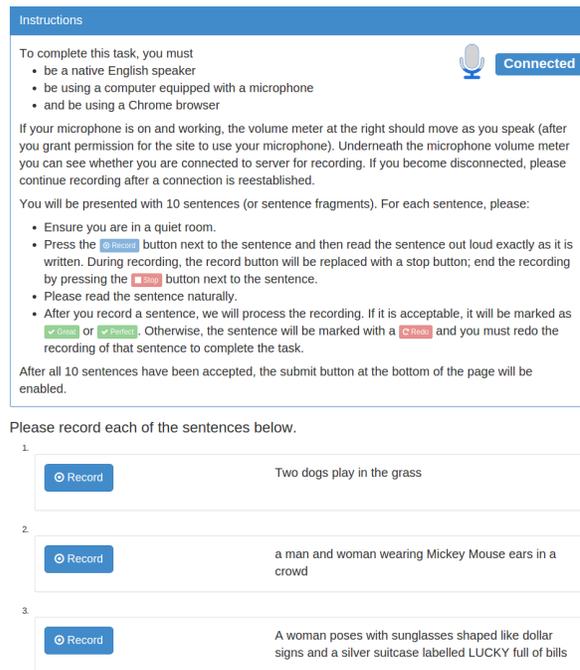


Fig. 3. Audio collection interface for capturing spoken captions on Amazon Mechanical Turk.

4. EXPERIMENTS

We use stochastic gradient descent with a learning rate of 1e-6 and momentum of 0.9 across batches of 40 images to train

our embedding and alignment model, and run our training for 20 epochs. Training is performed using the standard 6,000 image train set from the Flickr8k data, using the accompanying 30,000 captions. At each batch, we randomly choose to use only one of the five captions associated with each image. We tried several different settings for h , the dimension of the semantic embedding space, and found that values between 512 and 1024 seemed to work well, in line with [8]. We also found that it was necessary to normalize the w vectors to unit magnitude in order to prevent exploding gradients.

To evaluate the alignment and embedding model, we follow the example of [6, 8, 21] and use our model to perform image retrieval and annotation. Image search is defined as choosing a caption from the test set and then asking the system to find which image belongs with the caption. Image annotation is the opposite problem: choosing an image from the test set without its caption, and then asking the system to search over all the captions in the test set and find one of the five which belongs with the image. We report recall@10 as our evaluation metric, or the probability that the correct result is found in the top 10 returned hits. Table 1 details the results of our system (“Spectrogram CNN”), as well as a comparison to replacing the word spectrogram embeddings with 200-dimensional word vectors taken from [22]. We also compare to Socher et al. [21] and Karpathy [8]. While our text + word vector system outperforms [8], the model is more similar to Karpathy’s refinements made in [6] but with a single layer word embedding network rather than a bidirectional recurrent neural network. [6] reports high recalls on the Flickr30k data (50.5 search and 61.4 annotation), but does not include any results on the Flickr8k data. Although our spectrogram CNN does not perform nearly as well as any of the systems with access to the ground truth text, it massively outperforms a random ranking scheme. This is in spite of the fact that not only does the spectrogram CNN system not have direct access to the ground truth word identity of the caption words, but also that the CNN word embedding vectors are of dimension 1024 rather than 200. We believe that these results are quite promising, and with more training data we expect to see substantial improvements. Figure 4 displays several alignments of Flickr8k images to their captions inferred by our system. While by no means perfect, our system reliably aligns salient objects in the images with their associated caption words.

We also trained several different word spectrogram CNNs with varying configurations. Table 2 displays the top-1 and top-5 accuracies of a few of these networks. A two-layer conventional DNN with 1024 units per layer and ReLU nonlinearities achieved a classification accuracy of 75.5%, while adding a third layer brought that number even lower to 69.5%. We speculate that our training set is not large enough to train such a network. However, replacing the first fully connected layer with a 64-unit convolutional layer (following the architecture described in Section 2.2) boosted the accuracy to 84.2%. We also trained a network with two convolutional

Model	Search R@10	Annotation R@10
Socher et al. [21]	28.6	29.0
Karpathy [8]	42.5	44.0
Text + word vec	49.0	56.7
Spectrogram CNN	17.9	24.3

Table 1. Image search and annotation results on the Flickr8k test images (1000 images with 5 captions each).

Model	Top-1 Acc.	Top-5 Acc.
DNN, 2x1024 FC	75.5	93.9
DNN, 3x1024 FC	69.5	91.4
CNN, 1x64 Conv + 2x1024 FC	84.2	97.4

Table 2. Isolated word recognition accuracies on our WSJ test set. “FC” stands for “fully connected”.

layers and one fully connected layer and achieved similar results to the network with only a single convolutional layer. We also explored varying the size and shapes of the convolutional filters, pooling layers, and dimension of the fully connected layers, but the network achieving 84.2% accuracy reflects our best performance. Although these networks show a wide range of top-1 accuracies, it is interesting to note that their top-5 accuracies are all in excess of 90%. Figure 2 displays the 64 filter responses from the first layer of our network.

5. CONCLUSION

In this paper, we have presented our first efforts to construct a model which can learn a joint semantic representation over spoken words as well as visual objects. At training time, the model only requires weak labels in the form of paired images and natural language spoken captions. Our system aligns salient visual objects in the images with their associated caption words, in the process building a semantic representation across both modalities. We evaluate our model on the Flickr8k image search and annotation tasks, and compare it to several systems with access to the ground truth text.

There are many avenues which we would like to take this research next. Deeper investigation of the performance gap between CNN speech embedding and ground truth text systems is a logical first step, and increasing the amount of training data may shed some light on this. We would also like to incorporate word level segmentation into the alignment scheme, alleviating the need to use forced alignment and making our setting more realistic. Lastly, while the neural networks used to extract features for both the visual objects and the spoken words are pre-trained in a supervised fashion on outside data, we believe that with a very large amount of data it may be possible to train them together along with the alignment and embedding model.



Fig. 4. Some examples of inferred alignments on the Flickr8k data. The words for each image’s caption are stacked to the right of each image, accompanied by their alignment scores. To keep the images free from too much clutter, we threshold the scores at 0, displaying a link between the word and its maximally associated object bounding box only when its score is positive. Note that the system does not actually see the text of the caption words - only a spectrogram. We replace the spectrogram in these figures with the ground truth text for the sake of clarity.

6. REFERENCES

- [1] K. Barnard, P. Duygulu, D. Forsyth, N. DeFreitas, D. M. Blei, and M. I. Jordan, “Matching words and pictures,” in *Journal of Machine Learning Research*, 2003.
- [2] R. Socher and L. Fei-Fei, “Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora,” in *Proceedings of the 2009 Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.
- [3] C. Kong, K. Lin, M. Bansal, R. Urtasun, and S. Fidler, “What are you talking about? text-to-image coreference,” in *Proceedings of the 2014 Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [4] D. Lin, S. Fidler, C. Kong, and R. Urtasun, “Visual semantic search: Retrieving videos via complex textual queries,” in *Proceedings of the 2014 Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [5] C. Matuszek, N. Fitzgerald, L. Zettlemoyer, L. Bo, and D. Fox, “A joint model of language and perception for grounded attribute learning,” in *Proceedings of the 2012 International Conference on Machine Learning (ICML)*, 2012.
- [6] A. Karpathy and L. Fei-Fei, “Deep visual-semantic alignments for generating image descriptions,” in *Proceedings of the 2015 Conference on Computer Vision and Pattern Recognition, CVPR*, 2015.
- [7] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, “Show and tell: A neural image caption generator,” in *Proceedings of the 2015 Conference on Computer Vision and Pattern Recognition, CVPR*, 2015.
- [8] A. Karpathy, A. Joulin, and L. Fei-Fei, “Deep fragment embeddings for bidirectional image sentence mapping,” in *Proceedings of the 2015 Conference on Computer Vision and Pattern Recognition, CVPR*, 2014.
- [9] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov, “Devise: A deep visual-semantic embedding model,” in *Proceedings of the Neural Information Processing Society*, 2013.
- [10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *Proceedings of the 2013 Conference on Computer Vision and Pattern Recognition, CVPR*, 2013.
- [11] J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders, “Selective search for object recognition,” in *International Journal of Computer Vision*, 2013.
- [12] S. Bengio and G. Heigold, “Word embeddings for speech recognition,” in *Proceedings of the 15th Conference of the International Speech Communication Association, Interspeech*, 2014.
- [13] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” in *arXiv preprint arXiv:1408.5093*, 2014.
- [14] C. Rashtchian, P. Young, M. Hodosh, and J. Hockenmaier, “Collecting image annotations using amazon’s mechanical turk,” in *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon’s Mechanical Turk*, 2010.
- [15] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, “From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions,” in *Transactions for the Association of Computational Linguistics*, 2014.
- [16] T. Y. Lin, M. Marie, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick, “Microsoft coco: Common objects in context,” in *arXiv preprint arXiv:1405.0312*, 2014.
- [17] P. Saylor, “Spoke: A framework for building speech-enabled websites,” M.S. thesis, Massachusetts Institute of Technology, 32 Vassar Street, Cambridge, MA 02139, 2015.
- [18] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, “The Kaldi speech recognition toolkit,” in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, 2011.
- [19] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large scale hierarchical image database,” in *Proceedings of the 2009 Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
- [20] D. B. Paul and J. M. Baker, “The design for the wall street journal-based csr corpus,” in *Proceedings of the Association for Computational Linguistics Workshop of Speech and Natural Language*, 1992.
- [21] R. Socher, A. Karpathy, Q. V. Le, C. D. Manning, and A. Y. Ng, “Grounded compositional semantics for finding and describing images with sentences,” in *Transactions of the Association for Computational Linguistics (TACL)*, 2014.
- [22] E. Huang, R. Socher, C. D. Manning, and A. Y. Ng, “Improving word representations via global context and

multiple word prototypes,” in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, 2012.