Identifying Complex Metadata Relationships to Objects using Natural Language Processing and Image Classification

Joel Lindsey[1], Andrew Pollock[2], Anand Rajan[1], Tej Tenmattam[1], and Benjamin Wilke[1]

[1] Master of Science in Data Science, Southern Methodist University, Dallas, TX 75275 USA {jdlindsey,anandr,ttenmattam,bwilke}@smu.edu
[2] Getty Images, New York, NY 10007 USA {Andrew.Pollock}@gettyimages.com
https://www.gettyimages.com/

Abstract. In this paper we present the integration of natural language processing and computer vision to identify complex relationships between objects in images. Computer vision has mostly been concerned with fairly low-level tasks, such as grouping contours, edges, and basic shapes in images to identify objects. Natural language processing is a much wider domain and includes tasks such as identifying syntax, semantics, and relations between words. We seek to combine these practices to express higher order relationships than simply identifying objects in an image. If a picture is worth a thousand words than our goal is to generate these words to not only understand the objects in an image, but interactions and activities between the objects. Our work will begin by mining keywords, captions, and other available descriptive narratives from images using natural language processing techniques. We will create a multimodal neural network combining convolutional and recurrent architectures to create a model that can maximize the likelihood of word combinations given a training image. Our research will also demonstrate how these encoder/decoder architectures can learn from only relevant interactions that occur in an image. Finally, we will test, document, and iterate over the results to maximize our potential for uncovering higher order classifications and provide quantitative and qualitative measurement of effectiveness.

1 Introduction

Semiotics is the study of sign process, which is the process where meanings are assigned to signs. Anything that communicates meaning to the interpreter of a sign, besides the sign itself, is considered in the process. The additional meaning can be intentional, as demonstrated in an ironic or comedic caption on a cartoon, or unintentional like a deodorant brand advertisement being displayed on a website that is not considered brand safe (containing explicit content). Signs can be communicated through the gamut of human senses including: tactile, visual, olfactory, taste and auditory. We will only be considering visual signs in the

form of digital images. It is our goal to explore semiotics as it can be understood through uncovering specific signs in image descriptions and narratives through natural language processing and apply them using neural network architectures.

Malik et al. [10] summarize the tasks associated with computer vision as reorganization, recognition, and reconstruction. Reorganization is related to "perceptual organization" in human vision, which is the bottom-up segmentation of image or vision components. For computer vision this is the low-level task of edge, contour, and corner detection performed by a kernel, or convolution matrix. These are passed to higher level tasks for semantic segmentation, which are recognition tasks for assigning labels to objects, scenes, events, and activities in an image. Recognition tasks are the most closely related to language as they apply meaning to objects with words. Computer and human vision share this connection to language by means of semantic representation. Finally, reconstruction is an estimation of a three-dimensional scene from one or more images by incorporating spatial layout, shading, textures, reflectance, and illumination. These tasks are the foundation for translation of low-level pixels, contours, and shapes to higher level descriptions of word combinations or whole sentences.

The authors of [12] suggest several applications for combining computer vision and natural language processing. Image captions can be generated in real time for news images or sub-titles for video content. Similarly, descriptions can be generated from medical images to streamline diagnoses by doctors. Images or video of sign language can be "translated" into speech or text. Finally, systems can be built to assist those with sensory disabilities, like providing blind people with spoken descriptions of their surroundings. The opposite conversion could produce images from spoken or written content to assist those who do not possess the ability of speaking or hearing.

In order to focus our project work we will only be exploring a specific set of object relationships. Simple object identification has become widely available through common trained image classification deep learning architectures - thus our goal will be to move to higher order relationships of the objects in an image. Our work will focus on descriptive adjectives about an object (a noun) itself, versus prepositions that describe proximity between objects (nouns). For example - we are interested in determining adjectives describing the size or color of a single object: "large cat", "blue ball", and "red sign" versus prepositions or prepositional phrases describing the relationships between two or more objects: "boy inside car", "grass outside window", or "statues on top of a building". We are also interested in preserving the correct context and adjective relationships when applying our labels. For example, a label associated with "soccer player wearing a blue jersey and red socks" should not be applied to an image with a person wearing a red jersey and blue socks. Our project sponsor Getty Images has identified the disambiguation of search terms as one the most challenging questions for image brokers, where market differentiation for their web presence is driven solely by search.

Our work will begin by identifying a limited set of object relationships to focus on the efficacy of our algorithm and model. We will leverage transfer learning

for our convolutional neural network encoder. Instead of applying long form descriptions along with the images our process will mine for specific combinations of parts of speech to focus our model training. Additional manual annotation of images may be required depending on the results of our text mining. Our multimodal neural network will consist of an LSTM decoder and it will be jointly trained with our convolutional encoder.

2 Related Work

In this section we provide relevant background on previous work on the task of image captioning requiring a computer vision system which localizes and describes important regions in images in natural language. Our work will build from two complementary papers on image caption generation [[15], [16]]. The task of image caption generation has been explored using solutions that attempt to merge the sub-tasks of object recognition and description, but it wasn't until recently that joint models optimized for these combined tasks emerged. The earliest inspiration was found in the advancement of machine translation. It was discovered that a model tasked with transforming a sentence S, into its translation T in the target language, could be achieved by maximizing $p(\mathrm{T/S})$. This discovery was an advancement from simpler approaches, which included naively translating words individually, without regard for surrounding words or context. The newest machine translation advancements take advantage of two-stage Recurrent Neural Networks (RNN) configured in an encoder/decoder architecture. The encoder RNN "reads" the input sentence to transform it into a fixed length vector representation. This vector is fed into the initial hidden state decoder RNN to generate a target output sentence. The proposal of [15] suggested that the encoder RNN could simply be swapped with a Convolutional Neural Network (CNN). The CNN would now generate an encoded vector representation of image contents to pass into the hidden layer of an RNN decoder to produce a "translation" - in this case an English description from an image. The challenge now is, how to efficiently train the encoder/decoder network as well as recognizing more complex interactions between objects in an image.

The authors of [16] innovated on this concept through the application of attention. CNNs are designed to benefit from reducing noise and clutter in an image to the most salient objects for simple object identification. However, the issue with applying these methods and compressing an entire image down to a single encoding is often interactions between objects will be lost. It's clear that a thoughtful algorithm should preserve this information by examining the low-level representations of an input image, however the new challenge is understanding how to steer the algorithm towards information that is relevant in the image. This concept was initially proposed in [2], which suggests that a fixed length vector that is produced by the encoder is actually a bottleneck. Bahdanau shows that compressing all the necessary information into a fixed length vector not only includes parts of the inputs that may not be important, but poses problems for inputs (sentences in this case) that are longer than any of the sentences

in the training corpus. "Bahdanau attention" - named for his contributions - most importantly proposes an encoder/decoder model that learns to align and translate jointly. That is, for each word generated in the decoder translation a soft-search is performed for areas of the source sentence where the most relevant related information can be found. The model uses context vectors associated with these areas and all the prior target predictions to inform the next word prediction. The authors of [16] demonstrate how Bahdanau attention can be applied to images.

3 Natural Language Processing Primer

Natural Language Processing (NLP) refers to the development and use of machine learning methods for processing and understanding natural language. Clearly machine learning models are not understanding language in the human sense, but rather they are trained to understand an imposed statistical structure of language [3]. Once the statistical structure of language has been learned it can be used to either understand new language inputs or produce new language outputs from arbitrarily trained inputs (other languages or in our case image representations).

Machine learning models must take arrays of numbers (vectors) as inputs, so we must preprocess our natural language data to be used in our model. Two simple approaches include one-hot encoding or index-encoding the vocabulary of words across our corpus. One-hot encoding creates a sparse vector for each word where the word is set to 1, while all other words are set to zero. This approach is very inefficient if we assume that even a modest vocabulary may have 10,000 words. This approach would leave 99.99% of the vector as zeros. Index-encoding involves assigning each unique word to a number and replacing each word in our corpus with a number, resulting in a dense vector each per example (where all elements are full). This approach suffers from additional deficiencies. The integer assignment is arbitrary, thus relationships between words are not captured. Furthermore, these integers are difficult for a model to interpret as a linear-classifier is assigned a single weight to each feature, but the input feature could vary greatly from one input vector to the next.

Word embeddings solve the aforementioned problems by providing a means to produce a dense feature representation, while also preserving word relationships. Most critically, these embeddings are not encoded by hand, but rather they are learned as part of the model training in the same way that weights are learned for any dense layer in a network. The word embedding length can be adjusted like other model hyperparameters and this decision is primarily influences by the size of the vocabulary. [1].

NLP can be used in text annotation, corpus analytics, sentiment analysis, search applications, machine translation and knowledge discovery. Human language is a complex hierarchy of concepts, including word meanings, etymology, grammar, inference, and social/cultural norms. This hierarchy is also represented

in the branches of research into NLP, including: lexical analysis, syntactic analysis, semantic analysis, and discourse/entailment analysis.

Syntactic analysis further involves:

- Sentence boundary detection
- Parts of speech tagging
- Parsing (shallow and deep)
- Lemmatization (for example if we parse 'great' as an adjective then we can determine its correct lemma).

Semantic analysis in NLP is divided into:

- Named entity extraction (NER)
- Relationship extraction
- Word sense disambiguation
- Classification (Tree, SVM)
- Tagging
- Segmentation of topic
- Sentiment analysis

Finally discourse analysis consists of:

- Anaphora resolution
- Discourse modeling
- Question answering
- Textual entailment
- Pragmatic analysis

4 Convolutional Neural Networks for Image Classification

Deep learning applications for image analysis have a long and rich history and significant advances in the field have been enabled by CNNs. CNNs comprise of basic building blocks such as convolutional layers, pooling layers, and fully connected layers.

When we train a CNN model on a large, diverse dataset, the layers within the model tend to learn filters or patterns that are relevant to the current task. For example, when trained on the task of image classification, we observe that CNNs learn hierarchical representations: early layers learn to detect simple patterns such as colors, lines, and edges, while later layers learn to detect complex patterns such as textures and parts of objects.

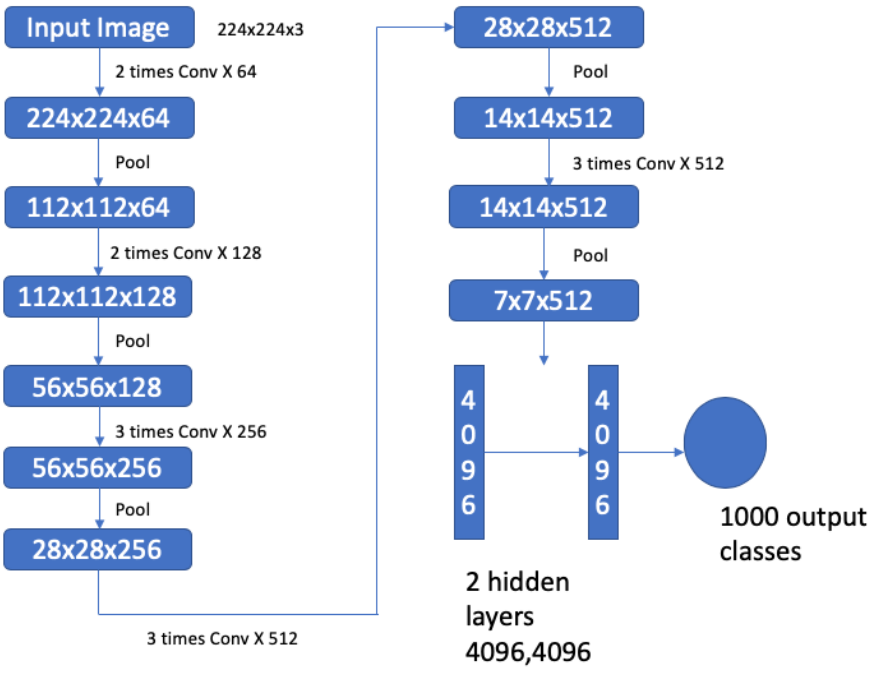
For our CNN pretrained model, we used the popular VGG-16 model, which has been trained on the ImageNet dataset.

4.1 Algorithm Primer and History

The VGG models for image classification come from the Oxford Visual Geometric Group. The VGG16 moniker is representative of a 16-layer model - there is also a VGG19 model with 19-layers. Both models included weights, in addition to the input and output layers, and a couple of max pooling layers. VGG-16 became renowned upon winning the 2014 ImageNet Large-Scale Visual Recognition Challenge (ILSVRC), which seeks innovation and accuracy in classifying a portion of the ImageNet dataset. The ImageNet dataset consists of 15 million labeled high-resolution images belonging to almost 22,000 categories, while the ILSVRC competition uses only a subset with roughly 1000 images in each of 1000 categories.

4.2 VGG-16 Architecture

VGG-16, was proposed by K. Simonyan and A. Zisserman from the University of Oxford in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition"[14].



Below are the steps performed in VGG-16 model:

– For an input image of dimension 224x224x3, a convolution layer of 64 is applied twice. The output will be such that the first two dimensions are the same, but the number of channels will be 64.

- Then a max pooling that divides this by 2 is applied, this gives us 112x112x64.
- Then a convolution of 128 is applied twice, which leaves the first 2 dimensions untouched and changes the third dimension to 112x112x128.
- Then a pool having a stride of two is applied, giving an output dimension of 56x56x128.
- This pool and convolution combination is applied until the first 2 dimensions turn 14.
- Then 3 convolutions of 512 are applied leaving everything untouched.
- The final step uses a pool to decrease the first 2 dimensions and leaves the number of channel untouched.

5 Transfer Learning

Transfer learning emerged from a curious phenomenon that lower layers of deep neural networks trained on natural images all resemble blobs and Gabor filters. These layers do not appear to be specific to any single classification task, but seem to generalize to all image classification tasks. Transfer learning suggests that lower layers trained on these general tasks can be reused from previously trained architectures. It's been shown that this practice will not only save computation, but may also improve generalization across disparate datasets [17]. Transfer learning can be implemented in several ways depending on project requirements. Earlier layers of a previously trained model are often deemed feature extractors as they are mostly concerned with low-level interactions between elements of an image, including: edges, contours, basic shapes, and color blobs. It is these layers that benefit the most from training on very large datasets like ImageNet. Modern CNN frameworks trained on ImageNet may take up to 3 weeks to train on cutting edge hardware - thus the benefits of leveraging existing pretrained CNNS for new applications is concerted.

The first application of transfer learning is to use a previously trained CNN as a fixed feature extractor. This is achieved by locking the feature extraction architecture from backpropagation and removing the later fully-connected layers where classification specific to a data set is occurring. These classification layers are replaced with a single fully-connected layer or with a more sophisticated series of fully-connected layers, perhaps implementing dropout for regularization. Retraining the model only involves learning the weights for the new classification architecture and the training can be done quite effectively with a very small amount of training data. The second approach to transfer learning is to initialize a pretrained network in an unlocked state and allow backpropagation to occur deeper into the pretrained network. This is known as "fine tuning" the pretrained network. This approach begins similar to transfer learning feature extraction. The new classification architecture must first be trained while freezing the feature extraction base network. If this is not done initially, the randomly initialized weights in the new classification layers will backpropagate large errors through the base network and obliterate the previously learned representation in the pretrained feature extractor. Once the new classification layer is trained some

of the layers of the base network can be unlocked and jointly trained with the entire network [3]. It's often beneficial to leave the lowest convolutional tiers locked from jointly training as abstraction at those tiers is fairly common across a breadth of image inputs.

The authors of [17] demonstrate that fine tuning approaches to transfer learning lead to improved generalization. They exhibit that even when a new dataset is not large, fine tuning an existing network will generalize better than training directly on the new dataset. Surprisingly, they are able demonstrate improvements in generalization when the existing network and new network are trained on disparate image domains. Finally, they are able to show that keeping any of the previous weights from the source network to initialize the second network results in improved performance (from only one layer to all).

6 Recurrent Neural Networks

Modern deep learning practitioners agree that the proposal of a deep neural network emulating the human brain is fairly absurd [3]. However, one thing remains clear, the human brain has capacity to remember short and long term inputs that are used to inform new inputs in real time. This concept diverges from a standard "feed forward" neural network architecture where inputs are converted to outputs by a fixed transformation. RNNs allow each cell in their architecture to feed the next cell alongside new inputs. A popular application of an RNN is predicting the next word in a sequence given not only the prior word, but some representation of a fixed number of all trailing words. Other RNN applications include time series forecasting for weather and financial market predictions.

All RNN architectures have the form of repeating modules. The simplest implementation proposed that inputs from the previous cell be combined with new inputs, activated via a tanh function, and then these outputs are passed both as the output of the cell and as inputs into the next cell.

**** INSERT GRAPHIC OF STANDARD RNN CELL *****

While the tanh activation function is required for this simple RNN it is plagued with the issue of the vanishing gradient. As small derivatives are allowed to backpropagate further into the network they become almost non-existent in the lower layers. This means that their weights will not be updated effectively with each training session, which is problematic since the lower layers are the building blocks for the entire network.

**** INSERT GRAPHIC OF LSTM RNN CELL ******

NEED MORE SPECIFICALLY ON LSTM

The architecture of RNN is that the input will have three dimensions, a batch size, number of steps and number of features. The RNN layer consists of single rolled layer that unrolls as we go through each step. RNN retains information from the previous step and feeds into the unrolled RNN units which are the hidden state. The current hidden state is calculated using the previous time

step's hidden state and current input step. All hidden state, output and input weights have same value throughout the RNN process.

RNN performs sequence labelling, text classification and text generation very well. These effectively use internal memory and based on previous inputs, predicts the ensuing output. The input vectors of each word sequence in the embedding matrix are the lookups. These are one hot encoded vector that corresponds to the word in the vocabulary. The input words go through the same embedding layer. The output usually is one hot vector that represents the word that follows or the sentence that is most close. The first hidden layer gets a first word vector and send the output activation to layer two. The second layer will take the second word and the activation output from first layer. The inputs are concatenated and sent to the third hidden layer if needed which does the activation similarly as the second one, until the last hidden layer. The output from this layer produces the one hot encoded vector of the word from vocabulary. This is called recurrent since, a looping iteration mechanism to retain the output through the process. This ensures that the iterations will have access to the previous predictions. For each input equal number of outputs are produced.

7 Multimodal Neural Model Integration

7.1 Dataset

Our training data will take advantage of the popular Microsoft COCO: Common Object in Context dataset (COCO) [9]. The COCO image dataset seeks to elevate simple object detection by placing objects in their natural context and encouraging complex scene understanding. To achieve these goals the COCO dataset provides non-iconic views of common objects in scenes comprised of many objects versus objects in isolation. The creators argue that other datasets often only include common (iconic) vantage points of objects and models trained on these images will struggle to recognize the same objects in other natural contexts. Ultimately their goal is to provide a dataset to advance research on contextual reasoning via computer vision related tasks. To achieve this goal images depicting scenes, rather than objects isolation, must be used.

Our testing data will be provided by our paper sponsor Getty Images.

7.2 Data Preparation

Our data preparation for image descriptions/annotations begins like most NLP projects. First, we coerce all letters to lowercase, remove special characters, and tokenize each word on spaces. We then add a special token to denote the beginning and ending of a sequence. This will be useful later to inform our decoder when a full phrase has been generated. We also replace words in descriptions with an "unknown" token for those words that appear only a few times in the vocabulary of all of our words. Finally, we create a dictionary that contains the mappings of our tokenized phrases back to the original words in the vocabulary.

7.3 Integration

Our goal is to maximize the probability of providing the correct image description, or simply accurate bigram pairs, by using this formulation:

$$\Theta^* = \arg\max_{\Theta} \sum_{(I,S)} log \Pr(S \mid I; \Theta)$$

where we optimize our parameters θ to maximize the log probability of S (the correct image description) given an image I and current parameters θ.

As S is a sentence and we are only interested in generating the next word this formula can be simplified as:

$$log \Pr(S \mid I) = \sum_{t=0}^{N} log \Pr(S_t \mid I, S_0, ..., S_{t_{-1}})$$

which calculates the probabilities for the whole description of length N for each next word S given an image I and each previous word S_t-1 to S_0.

7.4 Testing and Evaluation

Description generation systems prove to be very difficult to evaluate as incorrect nuance in written and spoken language is difficult to quantify without human intervention. The authors of [6] discuss some of the most prevalent errors found in the outputs of their multimodal neural model:

1. **Hallucination** is identifying an object that doesn't even exist in the image
2. **Counting** issues involve misrepresenting the amount of objects in an image
3. **Contradiction** occurs when multiple descriptions about the same object are not aligned
4. **Gender** issues are very prevalent and often pronouns are incorrectly identified or contradict in the same description generation
5. **Nonsensical** results are very common, and include incorrect replacement of nouns and verbs

A popular approach to evaluating description generation was proposed by Kishore Papineni, et al. [11] as the Bilingual Evaluation Understudy Score (BLEU). N-grams are defined as an n length sequence of contiguous words sampled from text or speech and they are commonly used in computational linguistics to decompose language for processing or analysis. The BLEU metric is generated by comparing n-gram candidates in the predicted description with n-grams in the reference text. These n-gram sequences are position independent and in general the more matches that are made between the candidate and reference texts the higher the BLEU score (a value between 0 and 1). The BLEU metric also takes into account that machine translation and image description generation systems tend to overgenerate reasonable words. The BLEU metric is designed to not reward candidate descriptions simply by the presence of reasonable words - what the authors [11] call modified n-gram precision.

8 Ethics

The promise of harnessing deep learning and transfer learning for image analysis is exciting and full of potential. However, there are important ethical issues that arise. In this section, we explore significant hidden traps related to biases in training data, privacy, and misinformation.

8.1 Training Dataset Bias

Deep learning is driven by the availability of large datasets used to train models. However the use of pretrained models for transfer learning depends on reusing others work. This means that anyone putting transfer learning into practice needs to be aware of the potential for bias in these models and take ownership of mitigating or documenting them. We are far from having the capability to produce bias-free models. For example, in the article "Machines Taught by Photos Learn a Sexist View of Women"[13], professor Vicente Ordóñez describes how he noticed a gender bias in the image recognition software he was building and how Microsoft's COCO dataset had significant gender bias. Data biases could also cause harm within other types of image data. For example, cancer lesion detection models built using medical image samples from people in North America will not work well for people in Asia and Africa. Self-driving car models trained on North American roads may struggle in parts of Asia and Africa. This could mean that life-saving cancer diagnosis software will be unavailable to certain parts of the world, or that the safety and efficiencies provided by autonomous vehicles won't be available to less-developed countries.

8.2 Privacy

To counteract the obvious harm that can be caused by surveillance using facial recognition a joint effort between policymakers and data practitioners needs to be created that respects privacy standards. Such policy, should meet the expectations of the public in terms of scope and fairness. Recognizing the lack of consent of the general public in general surveillance, the City of San Francisco in May 2019 became the first US city to ban public use of facial recognition [4]. However, the Chinese, Indian and UK governments [8] are moving in the opposite direction with regard to views on privacy in public image recognition.

8.3 Misinformation

Deep learning can now be used for image generation and natural language generation. This is great news for creativity and entertainment purposes, but it can also be used intentionally or accidentally to confuse or mislead people, or outright manufacture facts. Images or videos created using the image generation approaches are known as "deepfakes"[5]. In a similar vein, OpenAI created GPT2 [7] that can generate coherent paragraphs of text. These advancements in

deep learning can cause significant harm through misinformation. The solution will be a combination of technical and societal solutions where we track as to where the information originated from, how it is made, and how it is gotten and conveyed by people.

9 Conclusions

References

1. Tensorflow core - word embeddings. https://www.tensorflow.org/tutorials/text/word_embeddings, accessed: 2019-11-04
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate (2014), http://arxiv.org/abs/1409.0473, cite arxiv:1409.0473Comment: Accepted at ICLR 2015 as oral presentation
3. Chollet, F.: Deep Learning with Python. Manning Publications Co., Greenwich, CT, USA, 1st edn. (2017)
4. Conger, K.: San francisco bans facial recognition technology https://www.nytimes.com/2019/05/14/us/facial-recognition-ban-san-francisco.html
5. Foley, J.: 8 deepfake examples that terrified the internet https://www.creativebloq.com/features/deepfake-examples
6. Kiros, R., Salakhutdinov, R., Zemel, R.S.: Unifying visual-semantic embeddings with multimodal neural language models. CoRR **abs/1411.2539** (2014), http://arxiv.org/abs/1411.2539
7. Lang, F.: Openai's gpt2 now writes scientific paper abstracts https://interestingengineering.com/openais-gpt2-now-writes-scientific-paper-abstracts
8. Laskhmanan, R.: China's new 500-megapixel 'super camera' can instantly recognize you in a crowd https://thenextweb.com/security/2019/09/30/chinas-new-500-megapixel-super-camera-can-instantly-recognize-you-in-a-crowd/
9. Lin, T., Maire, M., Belongie, S.J., Bourdev, L.D., Girshick, R.B., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: common objects in context. CoRR **abs/1405.0312** (2014), http://arxiv.org/abs/1405.0312
10. Malik, J., Arbeláez, P., Carreira, J.a., Fragkiadaki, K., Girshick, R., Gkioxari, G., Gupta, S., Hariharan, B., Kar, A., Tulsiani, S.: The three r's of computer vision. Pattern Recogn. Lett. **72**(C), 4–14 (Mar 2016). https://doi.org/10.1016/j.patrec.2016.01.019, https://doi.org/10.1016/j.patrec.2016.01.019
11. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: A method for automatic evaluation of machine translation pp. 311–318 (2002). https://doi.org/10.3115/1073083.1073135, https://doi.org/10.3115/1073083.1073135
12. Shukla, D., Desai, A.A.: Integrating computer vision and natural language processing : Issues and challenges. VNSGU Journal of Science and Technology **Vol. 4**, 190–196 (2015)
13. Simonite, T.: Machines taught by photos learn a sexist view of women https://www.wired.com/story/machines-taught-by-photos-learn-a-sexist-view-of-women/
14. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv 1409.1556 (09 2014)

15. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: A neural image caption generator. CoRR **abs/1411.4555** (2014), http://arxiv.org/abs/1411.4555
16. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A.C., Salakhutdinov, R., Zemel, R.S., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. CoRR **abs/1502.03044** (2015), http://arxiv.org/abs/1502.03044
17. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? pp. 3320–3328 (2014), http://papers.nips.cc/paper/5347-how-transferable-are-features-in-deep-neural-networks.pdf