Multi-Modal Classification Using Images and Text

Stuart Miller[1], Justin Howard[1], Paul Adams[1], Mel Schwan[1], Robert Slater[1]
[1]Southern Methodist University, 6425 Boaz Lane,
Dallas, TX 75275 USA
{stuart, justinhoward, paula, mschwan, rslater}@smu.edu

Abstract. This paper proposes a method for the integration of natural language understanding in image classification to improve classification accuracy by making use of associated metadata. Traditionally, only image features have been used in the classification process; however, metadata accompanies images from many sources. This study implemented a multi-modal image classification model that combines convolutional methods with natural language understanding of descriptions, titles, and tags to improve image classification. The novelty of this approach was to learn from additional external features associated with the images using natural language understanding with transfer learning. It was found that the combination of ResNet-50 image feature extraction and Universal Sentence Encoder embeddings yielded a Top 5 error rate of XX% and Top 1 error rate of XX%, which is an improvement of XX on state-of-the-art results. This suggests external features should be used to aid image classification when external features are available.

1 Introduction

The performance of image classification methods has improved dramatically over the past decade, primarily due to advances in deep learning. Progress in the realm of computer vision has centered on deepening models (more layers) (He et al. 2015). More efficient architectures have made better use of the spatial characteristics of images (Simonyan et al., 2014; Szegedy et al, 2014). Ioffe and Szegedy (2015) introduced statistical methods to take advantage of the distribution of values within convolutional layers.

In parallel, Natural Language Understanding (NLU) has seen considerable advancement with the emergence of large corpora, models that retain sequence information over larger spans of text, and methods that leverage deeper lexical and semantic representations (Cer et al., 2018; Tai et al. 2015). Language learning models have evolved from the analysis of co-occurrences of words to word embeddings based on positional information gained through the analysis of encyclopedic volumes of corpora (Mikolov et al., 2013). Sequence models, such as recurrent neural networks (RNN) (Cleeremans et al., 1989) were used to extract syntactic information from word embedding sequences. Sequence models were improved by increasing model memory with long short-term memory (LSTM) (Hochreiter et al., 1997) networks, which combined multiple weights and activations to add a cell state capable of carrying forward more context. The current state of the art involves attention

mechanisms (Bahdanau et al., 2014; Vaswani et al., 2017), which is all that is needed to both encode and decode long- term contextual relationships between sequences of words.

To a large extent, these two fields have developed separately with image processing leveraging deep convolutional networks (Krizhevsky et al., 2012) and NLU using deep sequence-based networks (Tai et al. 2015). However, with high quality, transferable models for image data and text data, interest in multi-modal deep learning (learning joint deep representations from disparate types of data) has increased.

Recent studies indicate that deep representations of image data and text data learned from exceptionally large datasets are transferable to new datasets (Goodfellow et al., 2017). Interest in multi-modal learning in the context of images and text has focused applications of joint representations and self-supervised training. Applications of joint image and text representations have been related to embedding images into a semantic text vector space or inferring text embeddings from a visual vector space. Embedding images into the semantic text vector space improves search-and-retrieval of images (Petal et al., 2018). Similarly, embedding text into the visual vector space has been shown to improve image caption generation (Frome et al., 2013). These experiments in joint representation learning indicate a strong relationship between these two modes of data. Self-supervised learning in this area has typically focused on learning to classify images from noisy labels. Li and associates (2017) showed that images from the web could be classified using web metadata. Noting the strong relationship between text representations and image representations, this study focused on leveraging joint representations of image and text to augment classification tasks.

Traditionally, image classification models have exclusively used features extracted from images only. While this is a reasonable approach for many tasks where images are provided in isolation, in many cases such as the web, images are accompanied with metadata. This raises a natural question: *Can image classification tasks be improved by using associated contextual data?*

This paper presents an architecture[1] for learning deep representations of images and text and shows that multi-model learning can be used to enhance image classification. To combine feature extraction from images and text, this model provides input for images and an input for associated metadata text. The images and text are initially processed in parallel towers of deep convolutional and sequence networks, respectively. The initial layers extract features specific to the data type. These features are flattened and concatenated into a single feature vector, grouping image features and text features separately. Finally, a Dense Neural Network (DNN) predicts the image class from the combined feature vector.

This paper presents a set of comprehensive experiments with this model architecture on the WebVision dataset (Li et al., 2017) to show how metadata inclusion affects image classification performance. The model presented in this paper provides a performance of XX% Top 5 accuracy, which is an increase of X over the baseline state-of-the-art model provided with WebVision.

[1] Code is available at https://github.com/WebVision-Capstone/WebVision-Cap

2 Related Work

Two concepts are fundamental to this study: image classification and natural language understanding. Since the success of AlexNet in 2012 (Krizhevsky et al., 2012), the application of convolutional neural network models in image processing have been a dominant area of research. Similarly, sequence neural network models have dominated recent research in NLU. This study combines these two areas of research, focusing on improving image classification models with joint learned representations with text. In addition, exceptionally large models are required to train modern neural network models. Datasets for image classification and the fundamentals of convolutional image classification models and sequence NLU models are described in the following sections.

2.1 Image Classification Datasets

ImageNet datasets remain a dominant method of achieving a baseline trained model for image classification. The datasets are based on the hierarchically clustered Natural Language Processing (NLP) database of words and synonyms, WordNet 3.0 (Deng et al., 2009). ImageNet currently draws only nouns from this established hierarchy of 117,000 unique synsets. ImageNet images are currently labeled according to one of the approximately 21,000 WordNet synsets, which means that each image classifier is severely limited (ImageNet, 2020).

An additional limitation of the ImageNet dataset is the time that it takes to update the data. Since the project's inception in 2012, 14 million images have been labeled and added to the ImageNet dataset compared to the 1.8 billion images uploaded to the internet each day. One of the greatest contributions to ImageNet's accuracy, and the time it takes to update the dataset, was the quality control process. Image labeling and the evaluation label accuracy was crowd sourced with Amazon's Mechanical Turk[2]. The labeling precision of 80 randomly sampled synsets of the original ImageNet DET dataset yielded an average of 99.7% accuracy (Deng et al., 2009). This suggested it was a very reliable source of high-quality data, which justified the cost to build the dataset.

The creators of the WebVision dataset showed that accurate image classification can be achieved using noisy images and the associated metadata taken directly from web searches (Li et al., 2017). The WebVision 2 dataset contains over 16 million images and their metadata, such as descriptions, titles, and tags (WebVision, 2020). The classification accuracy of models trained on the WebVision dataset offer comparable accuracy, and in some cases higher accuracy, to models trained using ImageNet, despite the presence of noise within the data. The creators of WebVision found that models that learn from web data differ from curated datasets in that they learned from the wide array of human annotations and captured the linguistic complexities of language more readily from metadata. Comparisons of models trained

[2] A marketplace for outsourcing virtual work; see https://www.mturk.com/

on WebVision to models trained on ImageNet showed the role that quantity can play in the accuracy of a model, despite the presence of noise.

2.2 Image Classification with Convolutional Neural Networks

In recent years, the use of CNNs led to significant progress in image classification tasks. This type of network is built from a set of layers designed to extract the salient spatial features within images. Early forms of CNNs like LeNet-5 (LeCun et al., 1989), essentially stacked pairs of two types of layers – 2D convolution and pooling. Convolutional layers are made of a set of square filters. Each filter is convolved over the input image, producing a smaller intermediate output image. Pooling layers down-sample the output images by splitting the input images into a square matrix and passing forward the maximum value or average value.

Two problems arise from deep stacks of these two types of layers. First, it is difficult to train very deep networks of this type because the gradient diminishes too rapidly during backpropagation, preventing the successful training of the most outer layers (Bengio et al, 1994). This is often called the vanishing gradient problem in the literature. Second, large networks are computationally expensive. In CNNs, the computational expense increases quadratically with a uniform increase in network size (Szegedy et al., 2014). Residual Networks (ResNet) proposed by He and associates (2015) were designed to mitigate the vanishing gradient problem. Inception networks were designed to improve the efficiency of convolutional layers by introducing sparsity into the convolutions (Szegedy et al., 2014).

This study employed both ResNet50V2 and Inception V3 as the CNN architectures for image classification. Additionally, transfer learning was exploited by using pre-trained weights for these models[3] (pretrained on ImageNet). ResNets, Inception layers, and transfer learning are described in the following sections.

ResNets. ResNets were designed to mitigate gradient loss in very deep convolutional neural networks. The central idea behind ResNet is the addition of an identity connection – a layer that skips one or more convolutional layers, passing the state of the previous layer around the convolution layer and summing with the output of the convolutional layer (He at al., 2015). With the addition of the identity mapping between sets of convolutional layers, the model learns residual mappings rather than learning the entire functional mapping. It was hypothesized by the authors that a residual mapping may be easier to learn than the total mapping (He at al., 2015). In the conceptional ResNet module shown in figure 1a, the input (X) is passed through a two-layer path (approximating F(X)) and a skip connection path. The outputs of the two paths are summed at the output of the ResNet. Feeding the identity of the input

[3] The pre-trained weights for many of these state-of-the-art classification models are made available in neural network programming frameworks such as TensorFlow (Abadi et. al, 2016). TensorFlow is a programming framework for neural network model development and deployment. See https://www.tensorflow.org/.

forward (by the skip connection), mitigates the vanishing gradient problem in very deep networks. The ResNet shown in figure 1b, which was used in ResNet-110 and ResNet-164 (He et al. 2016), is a more typical application of a ResNet module.

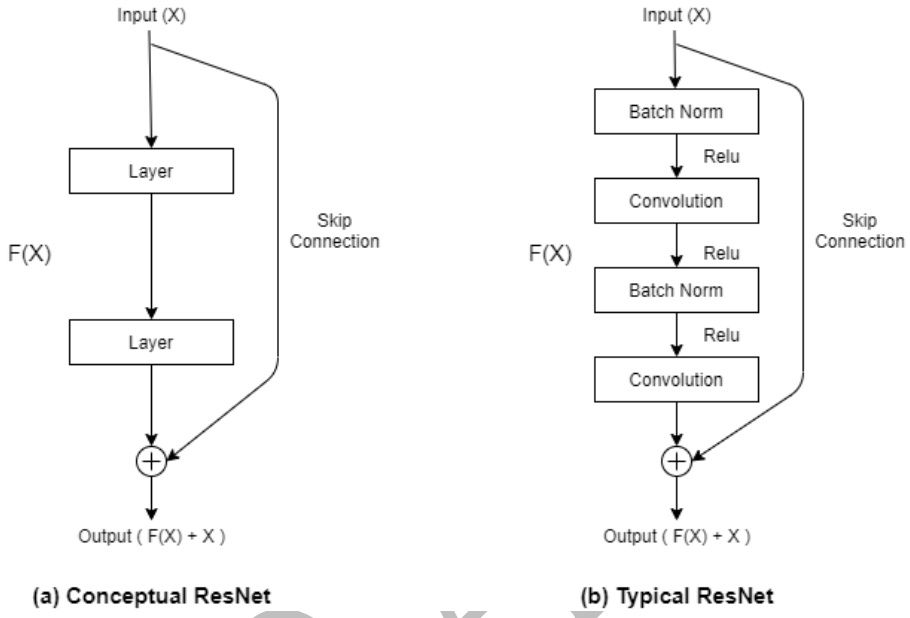

Fig. 1. ResNet building blocks. A conceptual ResNet module is shown in (a). A typical ResNet module is shown in (b).

Inception. Inception networks [4] were designed to improve the efficiency of convolutional layers by introducing sparsity into the convolutions. The inception architecture is based on the idea that the output of a given layer should be constructed so that correlated outputs are grouped together, which was suggested by Arora and associates (2013). The authors surmised that there should be clusters that are tightly packed as well as larger, more spread out clusters (Szegedy et al. 2014). The inception layer addresses this by performing three separate sets of convolutions with different sizes over the input and concatenating the resulting sets of filters as the output. The inception layers were used in place of the typical convolutional and pooling layers in the Inception V1 (GoogLeNet) model architecture. The layout of the original inception module is shown in figure 2. The primary sections of the module framed with solid borders. These primary layers perform feature extractions with three window sizes, 1x1, 3x3, and 5x5, to extract features of multiple sizes from the input images. The resulting tensors are concatenated together and passed to the next

[4] Inception networks are sometimes referred to as GoogLeNets in the literature, which comes from the author's team name in the ImageNet Large-Scale Visual Recognition Challenge 2014 (ILSVRC14) competition (Szegedy et al., 2014).

inception model. The 1x1 convolutional layers framed with dashed lines were inserted for dimensionality reduction. There have been several improvements to the original Inception architecture with ResNets added most recently in Inception V4 (Szegedy et al. 2016).

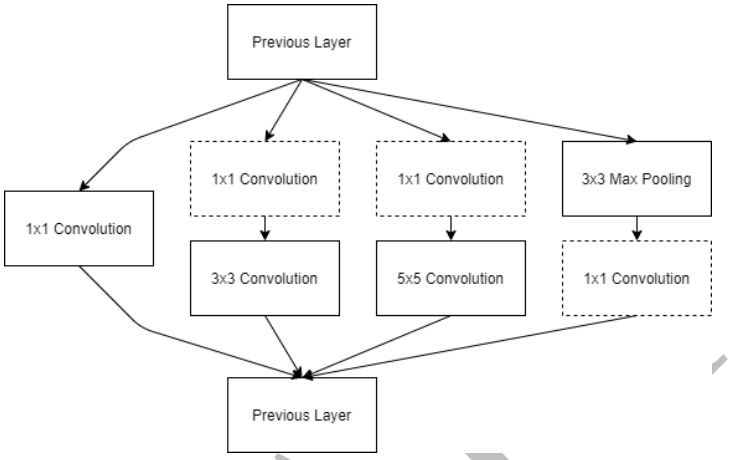

Fig. 2. Layout of Inception module from Inception V1 (GoogLeNet)

Transfer Learning on Images. The concept of transfer learning can be understood as applying a learner trained on a given task to a new task. In the context of deep learning, the representations learned in the initial layers from a task T1 may generalize to another task T2; thus, allowing the learner to be trained for T2 using very few examples (Goodfellow et al., 2017b). Yosinski and associates (2014) found that representations learned from training a CNN on images associated with nature related synsets could be applied to classifying images associated with man-made related synsets with little training. This suggests that features learned by features learned in the early convolutional layers have similar distributions to features that would have been learned from other images.

In practice, transfer learning on images is typically accomplished by replacing the last few layers (nearest the output) with layers for the specific problem (Pointer, 2019). This can be as simple as changing the output layer if the number of classes in the new task is different. Once the new layers are added, only the new layers of the model are trained for the new task, which decreases training time and the number of required training examples substantially (Pointer, 2019).

2.3 Natural Language Understanding

Natural Language Understanding (NLU) is the subset of Natural Language Processing - the other subset being Natural Language Generation - that deals with understanding input syntax, semantics, pragmatics, and discourse (Bates, 1995). Traditionally, this topic has been approached through statistical methods. However, deep learning has

risen to the forefront of NLU, which relies on natural language embedded into numeric vectors that can be used for natural language processing tasks with sequence models and transformer models (Cer et al, 2018). Methods for word embeddings, NLU sequence modeling, and transfer learning are discussed in the following sections.

Word2Vec. The Word2Vec model is a two-layer neural network that was created to encode and embed words into numeric vectors that can be used for arithmetic operation. Word2Vec operates on two basic models: the Continuous Bag-of-Words (CBOW) and the Continuous Skip-Gram. The CBOW model uses a continuous, distributed representation of the verbal context to predict the value of the current word while the Continuous Skip-Gram model predicts the verbal context using the current word (Mikolov et al, 2013).

GloVe. The Global Vectors (GloVe) model embeds words into distributed, numeric vectors useful for arithmetic operation. The word vectors are then processed into a global, log-bilinear regression model to leverage global matrix factorization and local context window methods (Pennington et al, 2014). Distances between words in co-occurrence matrices create word vector spaces that enable regression tasks to be applied to non-zero values therein.

Sequence Neural Network Models for NLU. The previous NLU methods are only vector representations of words or documents. While these types of representations encode lexical and semantic properties, the syntactic properties are generally not encoded by these methods. Sequence neural network models[5] are used to extract syntactic information from sequences of word vectors (Goodfellow et al., 2017a), which are fundamental to the primary NLU models used in this study.

The most basic type of sequence model is the unidirectional sequence model (shown in figure 3a). In this type of model, word vectors are sequentially concatenated with a learned hidden state and passed through a layer generating a new hidden state (Goodfellow et al., 2017a). This process is continued recursively until the end of the vector sequence. Depending on the use case, the output of the model is the last hidden state vector (single vector representation) or the series of hidden state vectors (sequence vector representation). The single vector representation is often used in NLU applications such as sentence classification and sentiment analysis.

In contrast, the sequence vector representation is often used in tasks such as tagging words within sentences with parts of speech. The primary weakness of the unidirectional sequence model is that contextual learnings are only carried in one direction (typically forward). This directional learning means that less learned context is available at the start of sentences and more learned context is available at the end of sentences (Goodfellow et al., 2017a).

[5] Common sequence neural network models used in practice are the recurrent neural network (RNN) cell (Cleeremans et al., 1989) and the long-shot term memory (LSTM) cell (Hochreiter et al., 1997).

The bidirectional sequence models were created to mitigate the unbalanced context learning of unidirectional models (shown in figure 3b). A bidirectional sequence model is essentially two unidirectional models where the word vector input sequence is reversed in one of the unidirectional models (Goodfellow et al., 2017a). Like the unidirectional model, the sequence vector representation or the single vector can be used as the model output. However, the output of a bidirectional sequence model is the concatenation of the outputs of the individual (forward and backward) sequence models (Goodfellow et al., 2017a).

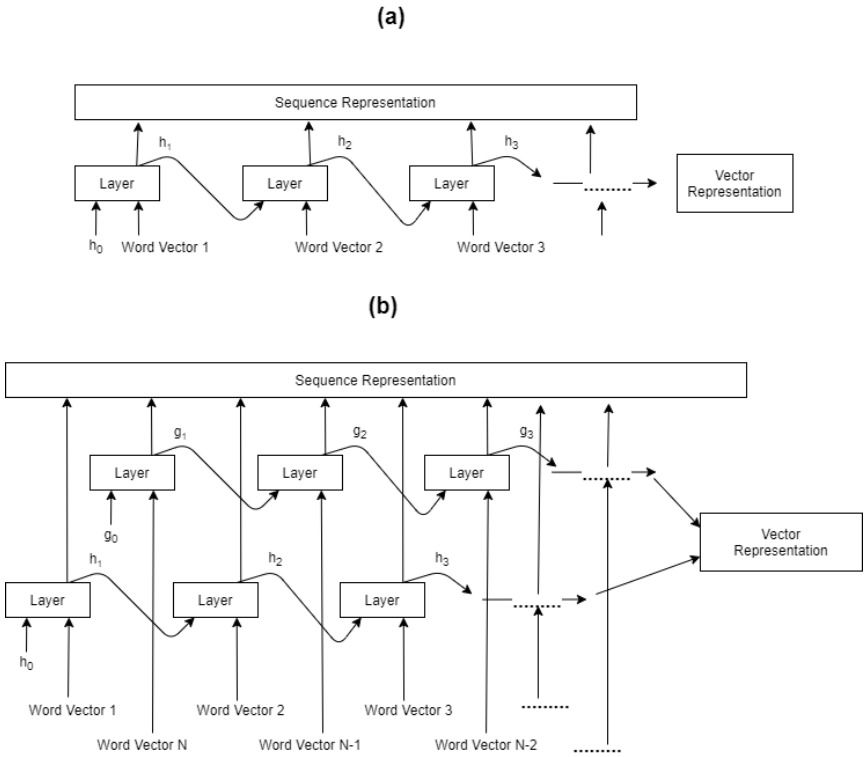

Fig. 3. Unidirectional sequence model (a) and bidirectional sequence model (b)

An additional sequence model is the encoder-decoder (shown in figure 4). This type of model is used to transform one type of sequence into another type of sequence, such as in language translation. A encoder-decoder consists of two parts: an encoder and a decoder. Either type of sequence model discussed previously can be used as the encoder (a unidirectional encoder is shown in figure 4), which encodes the entire sequence into a single vector representation (Sutskever et al., 2014). This single vector is passed to a decoder that autoregressively generates a new vector sequence from an internal hidden state and a start vector until an end vector is produced (Sutskever et al., 2014). The start and end vectors are learned representations

designed as signals to the neural network where a sentence starts and ends, respectively.

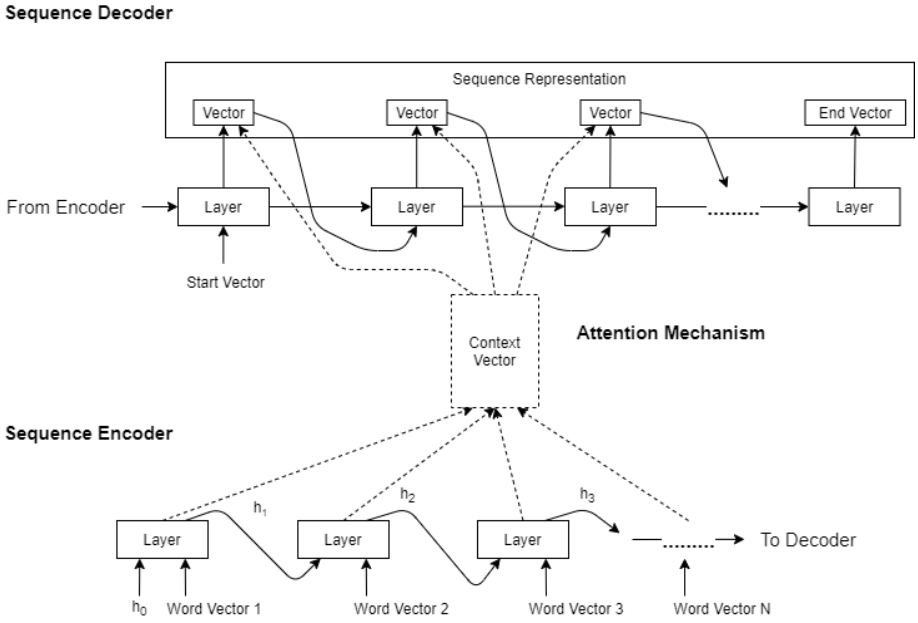

Fig. 4. An encoder-decoder sequence model with a unidirectional input sequence model and an attention mechanism.

The encoder-decoder sequence model had two main weaknesses: model performance drops off with longer sequences and sequence-based neural networks are expensive to train because training cannot be parallelized. These two weaknesses were addressed with attention and positional encoding. First, attention is a mechanism by which a neural network can leverage various parts of the input sequence while decoding the sequence (Bahdanau et al., 2014; Vaswani et al., 2017). In practice, attention substantially reduces sequence decoding error in long sequences (Bahdanau et al., 2014). Second, positional encodings eliminate the need to feed a sequence to the neural network, which enables parallelism of the input data (Vaswani et al., 2017). These two concepts are the main building blocks that make up the Transformer architecture, which is the basis of the Universal Sentence Encoder (USE) and Bidirectional Encoder Representations from Transformers (BERT).

Universal Sentence Encoder. As mentioned in Section 2.2, transfer learning is a practice whereby quasi-collinearity between at least two distributions enables a pipeline for information sharing from one distribution into the next. The Universal

Sentence Encoder[6] (USE) is applied to encode sentences into embedding vectors that can then be used for transfer learning. There are two models used for USE tasks: the Transformer model (Vaswani et al., 2017) – producing higher quality – and the Deep Averaging Network (DAN) (Iyyer et al., 2015) – providing shorter computation time.

The transformer-based approach constructs sentence embeddings using encoding sub-graphs, which compute context-aware representations of words in sentences (Cer et al, 2018). In a DAN, input embeddings for words and bigrams are averaged together then passed into a feed-forward Deep Neural Network (DNN), which produces sentence embeddings (Cer et al, 2018). Further processing for classification tasks following vector embedding with either the transformer-based approach or the DAN approach can be carried out within a DNN.

Bidirectional Encoder Representations from Transformers. BERT[7] is designed as a pre-trained sentence encoder. BERT is a deep bidirectional representation from unlabeled text which jointly conditions on both left and right context in all layers (Devlin et al., 2018). Fine-tuning can occur on the BERT model by adding one additional output layer to create models for a wide range of tasks. This project uses the transformer's attention mechanism to learn contextual relationships. Transformer consists of an encoder to read the text input and a decoder to produce a prediction for the task. BERT's goal is to generate a language model, and it only needs the encoder part. A series of tokens are the input for the BERT encoder, which are first converted into vectors and processed in the neural network. BERT adds metadata before it starts processing (Devlin et al., 2018).

The BERT architecture involves the preprocessing of text and the insertion of additional positional tokens. These tokens mark the beginning and end of paired sentences. Pairing sentences permits a greater contextual learning and ties sentences together. [CLS] tokens indicate the beginning of the first sentence, and [SEP] tokens separate the two sentences. Segment embeddings contain semantic data relating to the meaning of a phrase within a sentence, which lead to a deeper comparison of the relationships between phrases in addition to individual words. Positional information is also captured by BERT encoders. Positional embeddings capture the co-occurrence of word sequences within sentence pairs. This type of information contextualizes the word embeddings.

Training of BERT is accomplished with two separate strategies. Masked LM (MLM) strategy places a mask over 15% of the word tokens (Devlin et al., 2018). The model then attempts to predict the original value of the mask. Next Sentence Prediction (NSP) strategy the model receives pairs of sentences as input and learns to predict if the second sentence in the pair is the subsequent sentence in the original document.

[6] USE model extracted from https://tfhub.dev/google/universal-sentence-encoder/4

[7] BERT model extracted from https://tfhub.dev/google/bert_uncased_L-12_H-768_A-12/1

2.4 Multi-Modal Modeling

Multi-Modal Modeling of images and text combines semantic knowledge extracted from text with knowledge of spatial structures extracted from images. Models of this type learn joint representations of images and text. These joint representations have been used to relate images and text to improve search-and-retrieval, classification, and self-supervised learning. Additionally, training data from the web has been shown to yield more generalizable models. This study was focused on using multi-modal data to augment image classification tasks.

Self-Supervised Learning. As an alternative to fully human-supervised algorithms, recently, there has recently been a growing interest in self-supervised or naturally-supervised. These approaches make use of non-visual signals, intrinsically correlated to images, as a form of supervision for visual feature learning (Gomez et al., 2019). The prevalence of websites with images and loosely-related human annotations provide a natural opportunity for self-supervised learning. This differs from previous image-text embedding methods in that the goal is to learn generic and discriminative features in a self-supervised fashion without making use of any annotated dataset (Gomez et al., 2018).

Generalizability of Learnings From the Web. Research has lately focused on joint image and text embeddings. Merging different kinds of data has motivated the possibilities of learning together from different kinds of data, which put more focus on the field of study where both general and applied research has been done. A Deep Visual-Semantic Embedding Model (DeViSE) (Frome et al., 2013) proposes a pipeline that, instead of learning to predict ImageNet classes, learns to infer the Word2Vec (Mikolov et al., 2013) representations of their labels. By exploiting distributional semantics of a text corpus of every word associated with an image provides inferences of previously unseen concepts in the training set. Semantically relevant predictions make this model valuable even when it makes errors. These errors are generalized to a class outside the labeled training set (Patel et al., 2018; Gomez et al., 2019).

Generic Visual-Linguistic Representation Learning. Advancements in transferable vision models and transferable language models have led to the development of architectures for learning generic representations of images and text. Two such architectures are VisualBERT (Li et al., 2019) and Visual-Linguistic BERT (VL-BERT) (Su et al., 2019). VisualBERT (Li et al., 2019) is a transformer-based model (Vaswani et al., 2017) that integrates BERT (Devlin et al., 2018) with object detection models and self-attention to associate parts of input images to parts of input text. The attention mechanism allows VisualBERT (Li et al., 2019) to learn generic joint representations that are transferable between visual-linguistic tasks such as captioning an image. Similarity, VL-BERT is a transformer-based model (Vaswani et al., 2017) that relates embedded features of input text and images with an attention mechanism (Su et al., 2019). This use of attention enables the input vectors to aggregrate useful infromation from other sections of the input sequences (Su et al., 2019).

3 Methods

The WebVision dataset (Li et al., 2017) is a collection of images with associated web metadata. This study adopted an ensemble modeling approach to make use of the multi-modal nature of the WebVision dataset (Li et al., 2017) to improve classification results. The following sections provide insight into the processes guiding the formation of the multi-modal model's architecture.

3.1 WebVision Data

This study intentionally uses noisy images and text from the WebVision training set and its associated metadata, while excluding the cleaner validation data. This omission serves the intent to evaluate the utility of state-of-the-art NLP tools, USE and BERT. Additionally, validating the model with images and metadata that share a similar noise distribution to the training data provides a better assessment of model performance on loosely supervised data. Validation and test sets were created by randomly sampling 4% of the training data on a per class basis and splitting the sampled data into two equally sized sets. This sampling methodology maintains the noise distribution and class imbalance for each set of data, training, validation, and test.

The WebVision dataset is composed of 14 million images and metadata collected from Flickr[8] and Google Image Search[9] based on queries developed from the ImageNet Large-Scale Visual Recognition Challenge 2012 (ILSVRC12) synsets (Li et al., 2017). The metadata provided with the images consists of titles and descriptions (Flickr also provided hashtags). Two example images with associated metadata are shown in figure 5. The synset label for the top image in figure 5 is "black-backed gull, great black-backed gull, cob, Larus marinus." Notably, the synset is well captured by the image description, but the image title appears devoid of directly useful information. The synset label for the bottom image in figure 5 is "trestle bridge." In this case, the target synset appears in both the title and description, but with many other words. The other words in the titles and descriptions essentially add noise to the data.

Only minimal preprocessing was applied to the data. Images were transformed from original sizes to 300x300 frames with 3 color channels. Additionally, the image tensor values were scaled to be bounded between 0 and 1. Since the USE does not require text preprocessing, no preprocessing steps were performed on the text in model variants incorporating the USE for text vectorization. However, text tokenization[10] was performed on the input text for model variants utilizing BERT.

[8] https://www.flickr.com/

[9] https://images.google.com/

[10] BERT tokenizer:
https://github.com/tensorflow/models/blob/master/official/nlp/bert/tokenization.py

Description: "... Great Black-backed Gull swallows fish"

Title: "Birds on the hunt, DuPage county, Illinois (mostly)"

Description: "Bass Coast Rail Trail: Former Railway Trestle Bridge over Bourne Creek, Kilcunda"

Title: "Former Railway Trestle Bridge over Bourne Creek, Kilcunda ..."

Fig. 5. Example images and metadata from synset n02041875 (*top*) and synset n04479939 (*bottom*)

3.2 Exploratory Analysis

Since the WebVision data was provided without cleaning, elements of noise and missing instances were expected. The amount of missing metadata is shown in table 1. A significant amount of metadata for the Flickr images is missing. It was expected that the model would learn to ignore missing data and only use the image for classification.

Table 1. Missing metadata attributes.

Source	Metadata Attribute	Total Missing	Missing Percentage
Flicker	Descriptions	2,647007	34.3 %
	Titles	86,417	1.1 %
Google	Descriptions	660,331	7.9 %
	Titles	0	0.0 %

As mentioned in section 3.1, the WebVision dataset was collected in a unsupervised manner from Flickr and Google Images. This unsupervised data collection can lead to significant noise in the collected data. A synset affected by collection noise is shown in figure 6. The synset shown in figure 6 is flash camera (n03358726), which is captured by example (a). The other four examples in figure 6 (b-e) are common modes of noise that appear in images collected for this synset. These images were collected by this synset query because properties of the cameras used to produce these image were listed in the associated metadata.

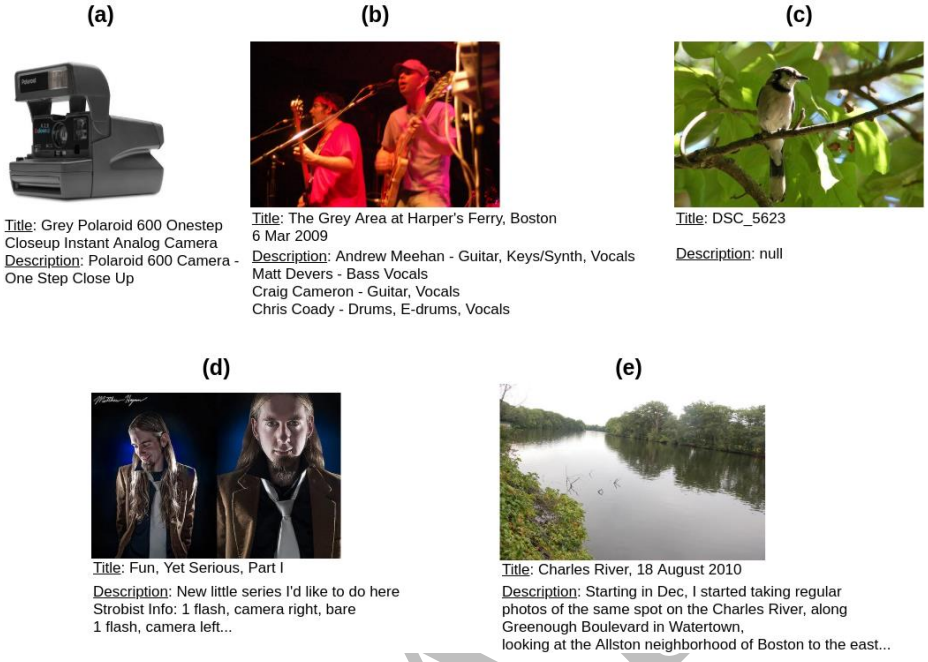

Fig. 6. Examples of noise modes in synset n03358726 (flash camera). Only one image (a) is correctly sorted into this synset. The other examples (b-e) are sorted into this synset but are not correctly labeled.

The BERT and USE layers transform text into vectors of size 768 and 512, respectively. These vector representations were transformed into a two-dimensional space using t-distributed Stochastic Neighbor Embedding [11] (t-SNE) for visual inspection of class separation (Kornblith et al., 2019; Pedregosa et al., 2011). The t-SNE embedding of the USE representations of descriptions and titles of 10 classes (selected at random from 5000 possibilities) are shown in figure 7 (top). Overall, the vectors do not appear to be well separated; however, the descriptions show more separation than the titles. The vector representations generated from BERT showed similar characteristics. Since fine-tuning BERT or the USE was not possible on the available hardware, an additional DNN layer was added between the output of the text vectorizer and concatenation to the image vector to provide pseudo-model tuning. The t-SNE of the learned representation after the DNN tuning layer showed better separation between the classes as shown in figure 7 (bottom).

[11] t-SNE was performed using the implementation provided in Scikit-Learn: https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html

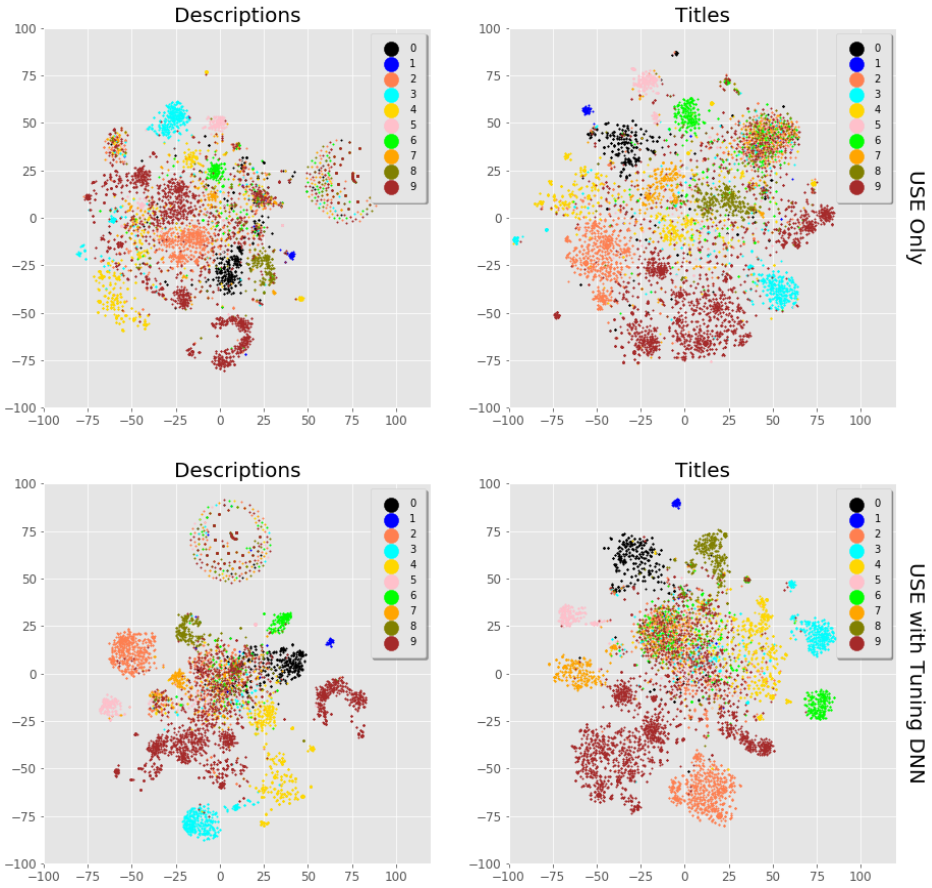

Fig. 7. Two-dimensional t-SNE embeddings of USE representations of metadata (top) and USE representations of metadata with a tuning DNN layer (bottom).

3.3 Model Development

This study combined multiple classification models to form a multi-modal image and natural language classification architecture. The distinct property of the model used in this study is the balance it achieves between the importance of image and metadata. Each image is accompanied by a title and a longer description. To establish a baseline comparison, the image and text classification models are validated separately. ResNet50V2[12], Inception V3[13], and MobileNetV2[14] image classification models were

[12] From: https://www.tensorflow.org/api_docs/python/tf/keras/applications/ResNet50V2
[13] From: https://www.tensorflow.org/api_docs/python/tf/keras/applications/inception_v3

trained solely on images. Likewise, USE and BERT models were trained to classify the titles and descriptions by their synset labels. [USE or BERT] was chosen for the final model based on the Top-5 Validation Accuracy[15]. Finally, multi-modal models that combine the image and metadata were trained to determine the effect of combining both methods of classification.

The image classification models apply the fixed features and fine-tuning methods of transfer learning. The first 50% of the layers of the ResNet50V2 and Inception V3 are frozen, while the last 50% of the layers are fine-tuned to achieve a balance of accuracy and training speed. The text classification models were not fine-tuned, however, the sentence embeddings produced by both the BERT and USE models were passed through DNN layers before being concatenated with the image embeddings. The concatenated multimodal representation of each image is then sent to a final classification layer. A conceptual diagram of this model is shown in figure 8. This model was implemented in TensorFlow (Abadi et. al, 2016).

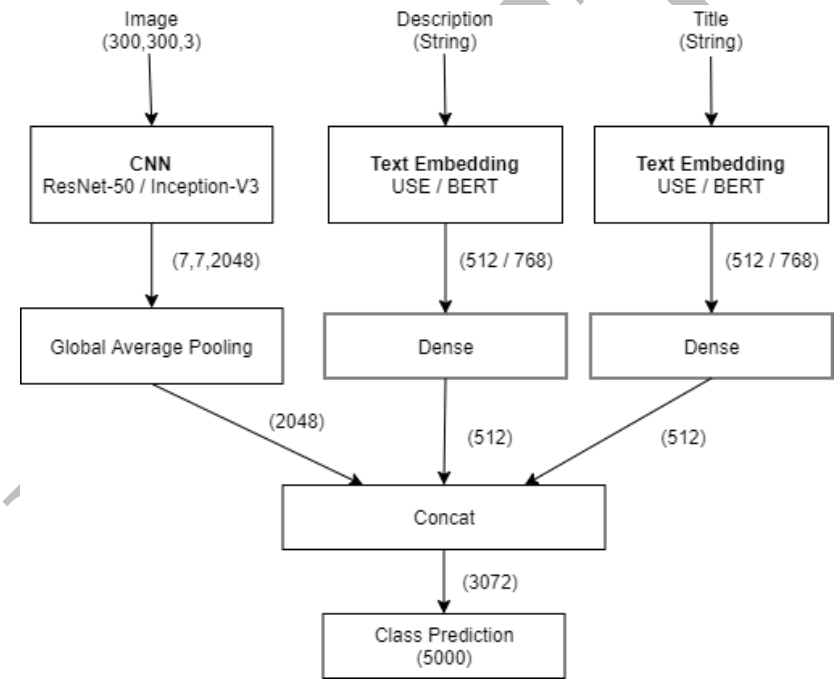

Fig. 8. Conceptual multi-modal model architecture combining image data and image metadata.

[14] From: https://www.tensorflow.org/api_docs/python/tf/keras/applications/MobileNetV2

[15] Top-5 Accuracy extends Top-1 Accuracy by counting an instance as correctly classified if the correct class is in the top 5 predicted probabilities. Top-1 Accuracy is the common definition of accuracy: the ratio of correctly classified instances to the total number of instances.

4 Results

Since this study involved two types of data, a baseline is provided for each type of data. An InceptionV3 model, a ResNet50V2 model and a MobileNetV2 model were used as baselines for image classification. USE and BERT text vectorizers with a DNN were used as baseline models for text classification. Six experiments were conducted using the proposed architecture: fine-tuned image feature extractor augmented with a pre-trained USE or BERT text vectorizer. The results of each model on the test set are shown in table 2. Overall, the models utilizing the image and text embeddings appear to perform better than the model using only a single mode of the data. The best model, X, provides an average improvement over the single-mode models of X.XX% and an increase of X.XX% over the model provided by the WebVision dataset creators (Li et al., 2017). The model results presented in table 2 are the performance of models on the holdout dataset discussed in section 3.

Table 2. Performance of models on WebVision test set.

Type	Model	Top-1 Accuracy	Top-5 Accuracy
Baselines	InceptionV3	44.50 %	67.38 %
	ResNet50V2	46.60 %	70.56 %
	MobileNetV2	44.39 %	68.22 %
	DNN-USE	50.30 %	66.21 %
	DNN -BERT	X %	X %
Experiment	InceptionV3-USE	X %	X %
	InceptionV3-BERT	X %	X %
	ResNet50V2-USE	53.58 %	71.78 %
	ResNet50V2-BERT	X %	X %
	MobileNetV2-USE	50.64 %	70.17 %
	MobileNetV2-BERT	X %	X %

5 Discussion

The exploration of multi-modal models presents unique advantages in terms of robustness to noise within the dataset and versatility. The following sections summarize the advantages of using multi-modal models with data collected using unsupervised processes.

5.1 Ensemble Advantages

Panel of Experts. The parallel ensemble resembles a “panel of experts” architecture. Each feature extraction tower acts as an expert and the concatenation of the extracted vectors acts as the panel. As shown in figure 10, each feature extractor separates the classes in different ways with different quality of separation.

The three trained towers together show an incremental improvement on the predictive power of the individual image classifier. Given that the text classification elements have a significantly higher prediction accuracy, they provide a needed balance that improves the predictive power of the image classifier by effectively denoising the image set.

A completely trained multi-modal model using this type of architecture could also be deconstructed to utilize the predictive capacity of the three parts that form the panel of experts. Separating the three models could provide semantic similarity metrics between classes based on sentence embeddings for the image titles, for example, and these similarity metrics can offer title or description calibrations for new or existing titles and descriptions.

Robustness to Missing Data. The ensemble architecture creates some robustness to missing data. When one input is missing, the other feature extractors still provide useful information for classification. The impact of missing data was analyzed with the following sets: five classes were selected at random from the body of data and the initial classification performance was evaluated. Then the model was tested on the same records, but with one item of data removed. The performance degradation from remove one of the three inputs was minimal.

5.2 Applications

There are a number of direct applications of this model architecture. Two direct internet-based applications are social media and image sharing websites. Images are a common medium used in social media websites. Social media images are typically accompanied with metadata entered by users. This type of model could be fine-tuned to automatically classify posts with images or generate vector representations of posts with images. This type of model could also be used by image sharing websites to classify images or create vector representations. As suggested by Gomez and associates (2019), deep joint representations of images and text can be used to improve search and query results.

The specific application of this multi-modal model allows a corporation to quickly tag images it hosts internally or from social media activity. An example of this activity would be the automatic generation of metadata for images hosted within websites. The top-5 most likely tags could be presented as suggested tags for uploaded media in an image-title-description format.

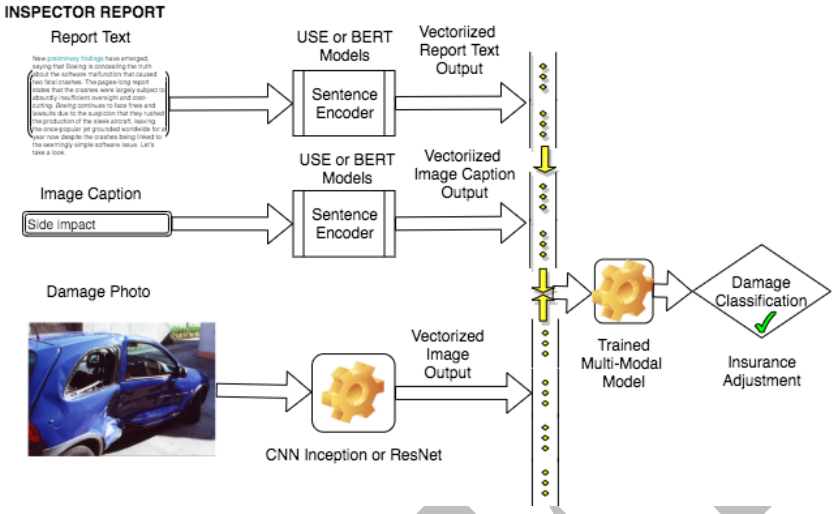

Fig. 9. Using multi-modal model for corporate social messages.

The possibilities for transfer learning of the features learned from the 5,000 classes of this model offer extensibility to the classification of far greater numbers of items. The features learned by the model could be used to implement additional fine-tuning to a quality check production line capacity. The multi-modal model could be given new sets of product defects, a brief summary of the type of defect, and a description of the ramifications or remedies for said defect. Queries of products could be associated with the images, titles, and descriptions, returning a much richer set of data.

A third application of the model would involve a machine-in-the-loop verification process, shown in figure 9. Image-title-description trios taken by humans, such as those that might be produced during an insurance claim, can be verified using the model in this study. Image-title-description trios can be classified by the claims inspector, verified by the model, then forwarded to a third party that verifies a correct classification of the image.

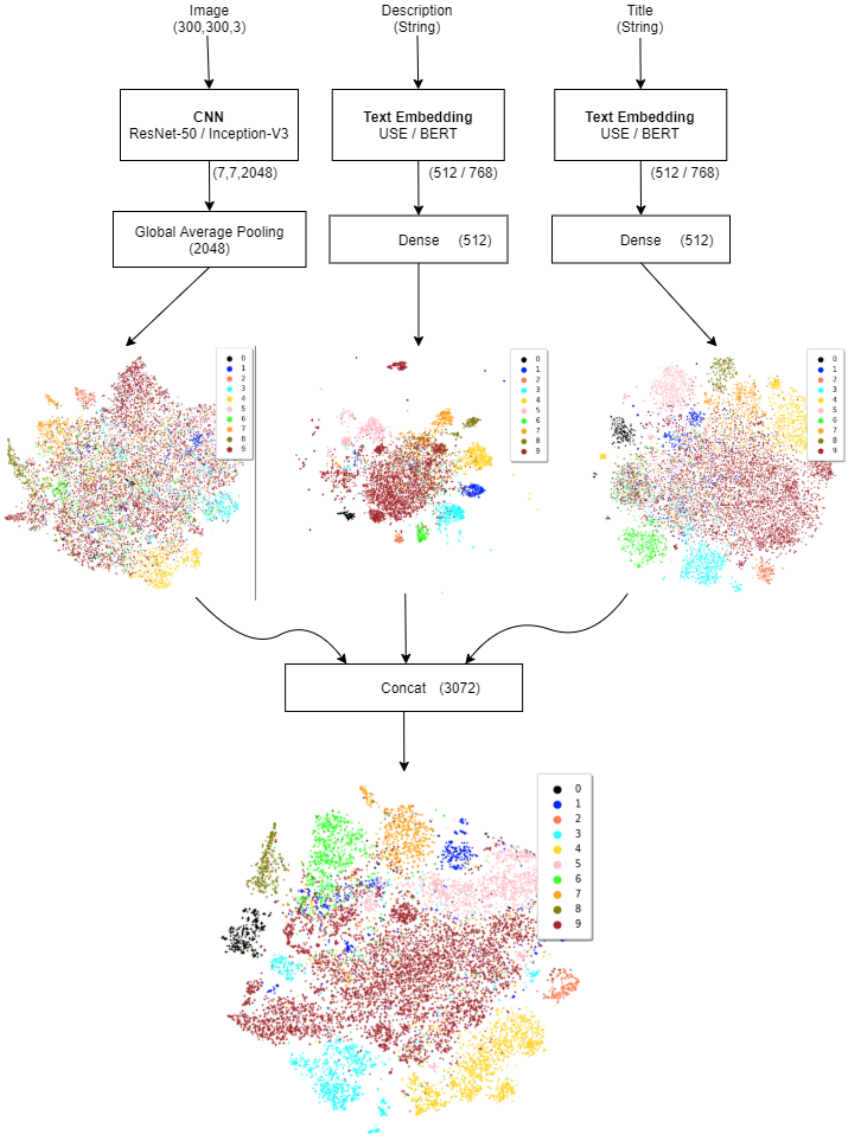

Fig. 10. Two-dimensional t-SNE embeddings of the vector representations of images from ResNet50V2 and vector representations of descriptions and titles produced by the USE.

5.3 Ethics

Algorithmic bias has been raised a serious issue with the growth of machine learning applications. Algorithmic bias as been shown to affect both computer vision models

and natural language models (Builamwini and Gebru, 2018; Bolukbasi et al., 2016). Since the WebVision datasets contains both types of data (images and text), bias contained within each mode of data could compound the effects on models. Additionally, the WebVision dataset contains noise within some classes, which may perturb the model learning process.

Dataset Bias. As noted in section 3, the WebVision dataset was collected in an unsupervised manner from image search engines. Since the collection process was unsupervised, the dataset inherited any biases present in the search engines or search engine results. Kay and associates showed that results from Google Image Search contained exaggerated gender stereotypes and unrepresented genders in certain careers (2015). Models trained on biased datasets may perpetuate learned biases. The effect of model bias on images was demonstrated by Builamwini and Gebru who showed that three commercial gender classification systems performed differently based on skin color (2018). Furthermore, Wang and associates showed models may amply biases existing in the dataset even for tasks not related to gender classification (2019).

This dataset text associated with the image data, which may also be a source of bias. Bias in word embeddings from top level algorithms such as GloVe (Pennington et al., 2014) and Word2Vec (Mikolov et al., 2013) has been well documented (Bolukbasi et al., 2016; Garg et al., 2018). However, the presence of bias in text embedding methods used in this study (BERT and USE) has not been deeply studied. Like GloVe and Word2Vec, dense vector representations of words are generated from BERT, but the representations from BERT are contextualized to the use case (Devlin et al., 2018). Kurita and associates demonstrated that BERT exhibited similar learned biases as GloVe and Word2Vec (2019). Unlike BERT, the USE does not create word embeddings, instead the USE generates vector representations from sentences (Devlin et al., 2018). Both the original authors of the USE (2019) and May and associates (2019) concluded that there is insufficient evidence to assert that the USE exhibits learned biases from text.

Data Collection Noise. As mentioned in section 3, some of the WebVision classes were perturbed with noise during the collection process. The vector representations of the images, descriptions, and titles of 10 classes produced at the concatenation layer of the model developed in this paper were mapped into a 2-dimensional vector space with t-SNE (Kornblith et al., 2019; Pedregosa et al., 2011) to visualize class separation and class relations (inspired by the work of A. Karpathy[16] and Gomez and associates) (2019). The t-SNE embedding and data examples of the 10 classes are shown figure 10. Based on figure 10, several classes such as “earwig” (n02272871) and “pea jacket, peacoat” (a03902756) appear well separated from other classes, while others such as “wrinkle, furrow, crease, crinkle, seam, line” (n013905792) and “flash camera” (n03358726) appear to exhibit more mixing with other classes. It is suspected that classes with generally good class separation were less affected by data collection noise. Classes that exhibit more mixing were either marred by data

[16] https://cs.stanford.edu/people/karpathy/cnnembed/

collection noise like “flash camera” as discussed in section 3.2 or are described in rather general terms like “wrinkle, furrow, crease, crinkle, seam, line”. Naturally, synsets of general terms will tend to capture a wider variance of items. Figure 11 shows a two-dimensional representation of noise instances in “flash camera” (n03358726).

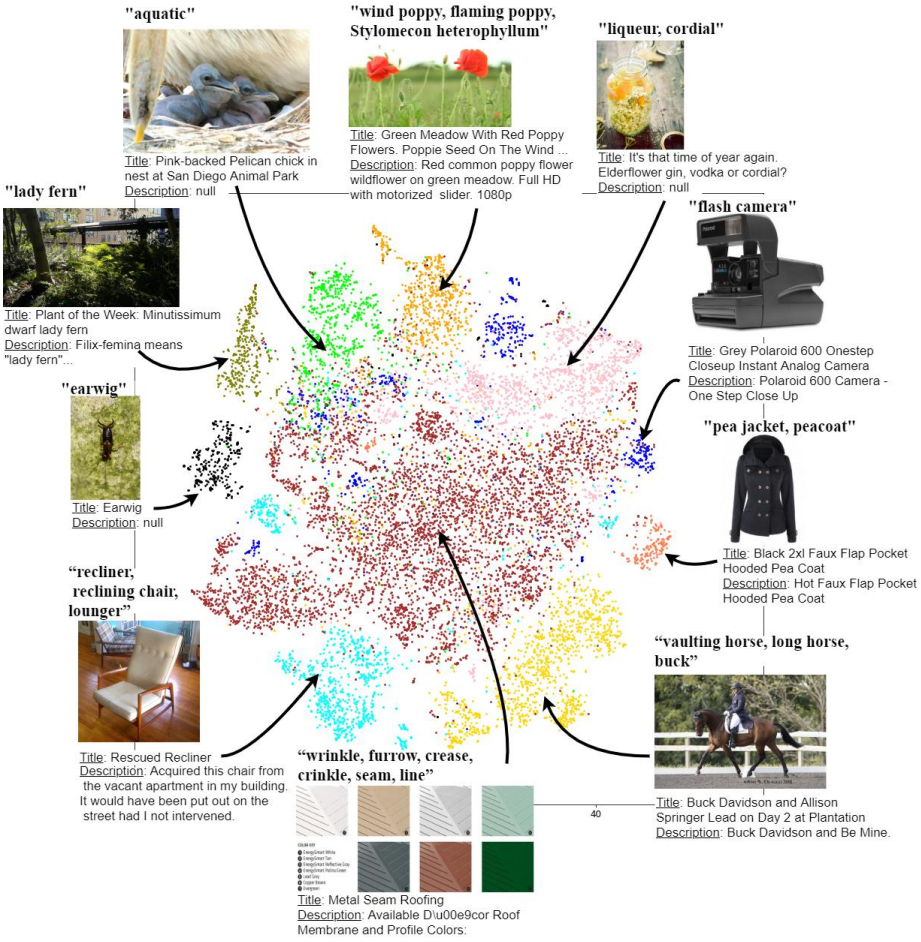

Fig. 11. Two-dimensional t-SNE embeddings of the vector representations of images, descriptions, and titles of 10 classes produced by the concatenation layer (last layer before classification) of the model developed in this study along with example instances.

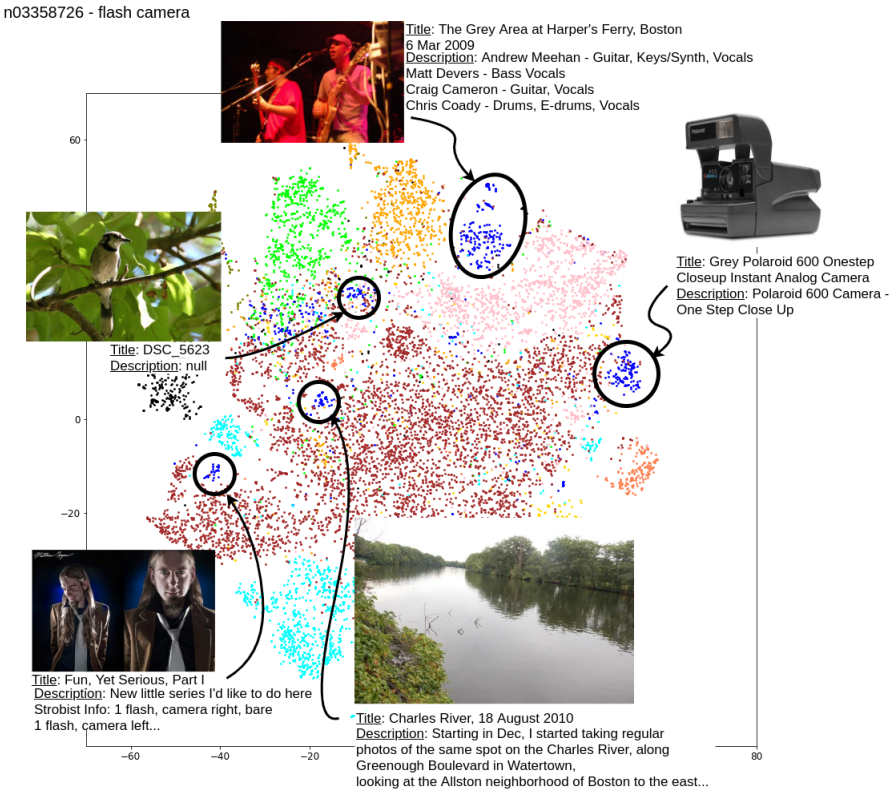

Fig. 12. Vector representations of the noise in synset n03358726 (flash camera) in the two-dimensional t-SNE vector space of images, descriptions, and titles produced by the concatenation layer (last layer before classification) in the model developed in this study. The same instances used to create figure 10 were used to generate this figure.

6 Conclusions

Continued improvements in image classification model development have progressed the realm of computer vision centered on deep learning. Approaches to enhancing deep learning models such as leveraging statistical methods to distribute spatial characteristics of images within convolutional layers have provided significant impact to this effort. Additionally, the advancement of deep learning tasks to solve Natural Language Processing problems using expansive lexical and semantic representations of language structures has been increasingly and reliably implemented for extracting meaning from vectorized character and word embeddings within dimensional space. Overlap in the foundational implementations of these two branched technologies has enabled the shared learning from each to impact the results of the other, in

collaboration. This paper asserts that transferred learning between these two approaches provides a robust solution to noise, improving the overall performance accuracy of classification tasks in which both media can be modeled.

Through this paper's comparison of baseline model performance – where classification tasks are performed separately for each medium – to the performance of experimental developments leveraging transferred learning between both media of baseline technologies proves the assertions that transfer learning between image and text classification enhances performance accuracy. In all models used, both top-1 and top-5 accuracy scores were more improved for the transfer-based models than the standalone (non-transfer based) models.

Respective of future developments, this paper will seek to produce continued statistical developments to further optimize transfer-based learning approaches to image and text classification.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., . . . Zheng, X. (2016). *TensorFlow: Large-scale Machine Learning on Heterogeneous Distributed Systems.* Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::af2da2320ce1d2a8a3fd36fa7ee3a861

"About ImageNet." *ImageNet*, Stanford Vision Lab, Stanford University, Princeton University. (2016). Retrieved from http://imagenet.stanford.edu/about-overview

Arora, S., Bhaskara, A., Ge, R., & Ma, T. (2013). *Provable Bounds for Learning Some Deep Representations* Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::96649dad9860ac80a3caf529b4081d5e

Bahdanau, D., Cho, K., & Bengio, Y. (2014). *Neural Machine Translation by Jointly Learning to Align and Translate*. Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::54d4837ebbe3b5ddd36de163e24a21cb

Bates, M. (1995). *Inception-learning*, Proc. Natl. Acad. Sci. USA Retrieved from https://www.pnas.org/content/pnas/92/22/9977.full.pdf

Bengio, Y., Simard, P., & Frasconi, P. (1994). *Learning long-term dependencies with gradient descent is difficult.* IEEE Transactions on Neural Networks, 5(2), 157-166. doi:10.1109/72.279181

Bolukbasi, T., Chang, K., Zou, J., Saligrama, V., & Kalai, A. (2016). Man is to computer programmer as woman is to homemaker? debiasing word embeddings. Retrieved from https://arxiv.org/abs/1607.06520

Buolamwini, J., & Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. *Proceedings of Machine Learning Research*, 81, 77-91. Retrieved from http://proceedings.mlr.press/v81/buolamwini18a/buolamwini18a.pdf

Cer, D., Yang, Y., Kong, S., Hua, N., Limtiaco, N., John, R. S., . . . Kurzweil, R. (2018). *Universal Sentence Encoder* Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::d6fe2dfd3083368f5f7b7a41b9dd3808

Cleeremans, A., Servan-Schreiber, D., & McClelland, J. L. (1989). Finite state automata and simple recurrent networks. *Neural Computation*, 1(3), 372-381. doi:10.1162/neco.1989.1.3.372

Deng, J., Dong, W., Socher, R., Li, L. Li, K., & Li, F. (Jun 2009*). ImageNet: A large-scale hierarchical image database.* Paper presented at the 248-255. doi:10.1109/CVPR.2009.5206848 Retrieved from https://ieeexplore.ieee.org/document/5206848

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::fa7dcff92cfaae2a75bbd8ef93e356f9

Fellbaum, Christiane (2005). WordNet and wordnets. In: Brown, Keith et al. (eds.), *Encyclopedia of Language and Linguistics*, Second Edition, Oxford: Elsevier, 665-670.

Frome, A., Corrado, G. S., Shlens, J., Bengio, S., Dean, J., Ranzato, M., & Mikolov. (2013). *T.DeViSE: A deep visual-semantic embedding model*.

Garg, N., Schiebinger, L., Jurafsky, D., & Zou, J. (2018). Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences - PNAS*, 115(16), E3635-E3644. doi:10.1073/pnas.1720347115

Gomez, R., Gomez, L., Gibert, J., & Karatzas, D. (2018). *Learning to learn from web data through deep semantic embeddings.* Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::270b8d6e6eee3a29b60259f915133dd5

Gomez, R., Gomez, L., Gibert, J., & Karatzas, D. (2019). *Self-supervised learning from web data for multimodal*. retrieval Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::5b7c79901802ec8f8278f36da67b7f89

Goodfellow, I., Bengio, Y., & Courville, A. (2017). Sequence modeling: Recurrent and recursive nets. In T. Dietterich (Ed.), *Deep learning* (pp. 363-408). Cambridge, MA: MIT Press.

Goodfellow, I., Bengio, Y., & Courville, A. (2017). Transfer Learning and Domain Adaptation. In T. Dietterich (Ed.), *Deep Learning* (pp. 526-531). Cambridge, MA: MIT Press.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). *Deep residual learning for image recognition*. Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::e7235b2295e7fd00c3555a8bfeb2c6b0

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Identity mappings in deep residual networks. Retrieved from https://arxiv.org/abs/1603.05027

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735. doi:10.1162/neco.1997.9.8.1735

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. Retrieved from https://arxiv.org/abs/1502.03167

Iyyer, M., Manjunatha, V., Boyd-Graber, J., & Daumé III, H. (2015). Deep unordered composition rivals syntactic methods for text classification. Paper presented at the doi:10.3115/v1/p15-1162 Retrieved from https://search.datacite.org/works/10.3115/v1/p15-1162

Kay, M., Matuszek, C., & Munson, S. (Apr 18, 2015). Unequal representation and gender stereotypes in image search results for occupations. Paper presented at the 3819-3828. doi:10.1145/2702123.2702520 Retrieved from http://dl.acm.org/citation.cfm?id=2702520

Kiros, R., Zhu, Y., Salakhutdinov, R., Zemel, R. S., Torralba, A., Urtasun, R., & Fidler, S. (2015). *Skip-thought vectors*. Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::446b9fed34d2cde1f12765f9158e81ca

Kornblith, Jonathan, Shlens, Jonathan, Le, V. Quoc. (2019). Do better ImageNet models transfer better? Retrieved from https://arxiv.org/abs/1805.08974

Krizhevsky, A., Sutskever, I., & Hinton, G. (2012, ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60, 84-90. doi:10.1145/3065386 Retrieved from http://dl.acm.org/citation.cfm?id=3065386

Kurita, K., Vyas, N., Pareek, A., Black, A. W., & Tsvetkov, Y. (2019). Measuring bias in contextualized word representations. Paper presented at the doi:10.18653/v1/w19-3823 Retrieved from https://search.datacite.org/works/10.18653/v1/w19-3823

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. & Jackel, L. D. (1989). *Backpropagation applied to handwritten zip code recognition.*

Li, W., Wang, L., Li, W., Agustsson, E., & Gool, L. (2017). *WebVision database: Visual learning and understanding from web data*. Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::33e2e9003bf050bf0f3bafaf378128a4

Li, L. H., Yatskar, M., Yin, D., Hsieh, C., & Chang, K. (2019). VisualBERT: A simple and performant baseline for vision and language. Retrieved from https://arxiv.org/abs/1908.03557

May, C., Wang, A., Bordia, S., Bowman, S. R., & Rudinger, R. (2019). On measuring social biases in sentence encoders. Retrieved from https://arxiv.org/abs/1903.10561

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013*). Efficient estimation of word representations in vector space.* Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::7d58576a90b92bef1fcc9e579134295f

Patel, Y., Gomez, L., Gomez, R., Rusiñol, M., Karatzas, D., & Jawahar, C. V. (2018). *TextTopicNet - self-supervised learning of visual features through embedding images on semantic text spaces.* Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::97b9755e269f4f875b8fbbf4c2afcc91

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Duchesnay, É. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, Retrieved from https://hal.inria.fr/hal-00650905

Pennington, J., Socher, R., & Manning, C. (2014). *Glove: Global vectors for word representation.* Empirical Methods in Natural Language Processing (EMNLP), 1532-1543. doi:10.3115/v1/D14-1162

Pointer, I. (2019). Transfer learning with ResNet. *Programming PyTorch for deep learning* (pp. 51-53). Sebastopol, CA: O'Reilly Media Inc.

Simonyan, K., & Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition*. Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::817f05f7aff35beeac308419b9b028c3

Su, W., Zhu, X., Cao, Y., Li, B., Lu, L., Wei, F., & Dai, J. (2019). VL-BERT: Pre-training of generic visual-linguistic representations. Retrieved from https://arxiv.org/abs/1908.08530

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). *Sequence to sequence learning with neural networks*. Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::ada95bfac7eb090942649e38bdabed8c

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., . . . Rabinovich, A. (2014). *Going deeper with convolutions*. Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::5df6d69237f08518896984361ca7485a

Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. (2016*). Inception-v4, inception-ResNet and the impact of residual connections on learning*. Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::44ac8f8822c35366d643221ef4b97e42

Tai, K. S., Socher, R., & Manning, C. D. (2015). *Improved semantic representations from tree-structured long short-term memory networks*. Retrieved from https://arxiv.org/abs/1503.00075

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., . . . Polosukhin, I. (2017). *Attention Is All You Need*. Retrieved from https://www.openaire.eu/search/publication?articleId=od_____18::4bcf8ae6d49b00d7d2e1624298c9764f

Wang, T., Zhao, J., Yatskar, M., Chang, K., & Ordonez, V. (Oct 2019). Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations. Paper presented at the 5309-5318. doi:10.1109/ICCV.2019.00541 Retrieved from https://ieeexplore.ieee.org/document/9008527

"WebVision Dataset 2.0." *WebVision: Visual Understanding by Learning from Web Data*, Computer Vision Laboratory, ETH, Zurich. (2018). Retrieved from data.vision.ee.ethz.ch/cvl/webvision//dataset2018.html.

Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). *How transferable are features in deep neural networks?*