DEEP SENTIMENT ANALYSIS ON TUMBLR

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose a novel approach to Sentiment Analysis using Deep Neural Networks combining Visual Recognition and Natural Language Processing. Our approach leverages Tumblr posts containing images and text to predict the emotional state of users. Deep convolutional layers extract relevant features from images and high-dimensional word embedding followed by a recurrent layer process the textual information in order to infer the emotion conveyed by a given Tumblr post. We demonstrate that our network architecture, named Deep Sentiment, learns meaningful relations between visual data and language as it vastly outperforms models using a single modality. We then show that Deep Sentiment can also be adapted to generate images and text representative of an emotion.

1 INTRODUCTION

Sentiment analysis has been an active area of research in the past few years, especially on the readily available Twitter data, e.g. Bollen et al. (2011) who investigated the impact of collective mood states on stock market or Flaxman & Kassam (2016) who analysed day-of-week population well-being.

Contrary to Twitter, Tumblr posts are not limited to 140 characters, allowing more expressiveness, and are not focused on the textual content but on the visual content. A Tumblr post will almost always be an image with some text accompanying the latter. Pictures have become prevalent on social media and characterising them could enable the understanding of billions of users.

We propose a novel method to uncover the emotional state of an individual posting on social media. The ground truth emotion will be extracted from the tags, considered as the 'self-reported' emotion of the user. Our model incorporates both text and image and we aim to 'read' them to be able to understand the emotional content they imply about the user. Concretely, the Deep Sentiment model associates the features learned by the two modalities as follows:

- We fine-tune a pre-trained Deep Convolutional Neural Network, named Inception (Szegedy et al., 2015), to our specific task of emotion inferring.
- We project the text in a rich high-dimensional space with a word representation learned by Word2Vec (Mikolov et al., 2013). The word vectors then go through a Recurrent Neural Network which preserves the word order and captures the semantics of human language.
- A fully-connected layer combines the information in the two modalities and a final softmax output layer gives the probability distribution of the emotional state of the user.

We will also see that Deep Sentiment can be rearranged to generate Tumblr posts expressing one of the learned emotion.

2 TUMBLR DATA

2.1 THE DATASET

Tumblr is a microblogging service where users post multimedia content with the following attributes: an image, text and tags. The tags are really valuable as they indicate the user's state of mind when writing his post. Ekman popularised the idea that there are six basic emotions [cite Ekman]: happiness, sadness, anger, surprise, fear and disgust. These emotions are said to be *basic*

as they are hardwired regardless of the species: basic emotions are innate, universal, automatic and induce fast reactions that are linked with a high survival rate. [Might need to rewrite when posts with other emotions are added]

To build our dataset, queries were made through the Tumblr API searching for each of the six emotions appearing in the tags. We considered adjectives as they were more commonly used on Tumblr: #happy, #sad, #angry, #surprised, #scared and #disgusted. The emotion would then become the *label* of the post that has visual and textual contents as features. Figure 1 shows two posts with their associated emotions:

(a) **Happy**: "Just relax with this amazing view #big-sur #california #roadtrip #usa #life #fitness (at McWay Falls)"

(b) **Sad**: "It's okay to be upset. It's okay to not always be happy. It's okay to cry. Never hide your emotions in fear of upsetting others or of being a bother. If you think no one will listen. Then I will."

Figure 1: Examples of Tumblr posts

2.2 Data preprocessing

In some posts, the tag containing the emotion of the post also appeared in the text itself, undermining the possibility of learning meaningful relationships between the features and the emotion expressed. We thus removed from the text the tags containing the emotion to be predicted.

Tumblr is used worldwide but we only kept posts written in English. Basically, if a post contained less than a given number of English word, it was deemed as non-English and removed from the dataset. The threshold was set to 5 English words as it appeared to filter out reasonably well non-English posts. [To change: filter posts with less than 90% of English words] The vocabulary of English words was obtained from Word2Vec.

[talk about dataset size, and reduction through preprocessing]

3 Visual recognition

Pictures are valuable to accurately infer the emotion expressed by a user. For instance, happy photos might contain sunny landscapes while sad pictures might contain darker colors. To extract visual insights from images, we will use convolutional neural networks, which achieve state-of-the-art performances in many visual recognition tasks.

3.1 Transfer learning

Training a convolutional network from scratch can be difficult as a large amount of data is needed and many different architectures need to be tested before achieving satisfying performances. To circumvent this issue, we can take advantage of the pre-trained network named Inception [cite] that learned to recognise images through the ImageNet dataset with a deep architecture of 22 layers.

Inception learned representations capturing the colors and arrangement of shapes of an image, which turn out to be relevant when dealing with images even for a different task. We could also say that

the pre-trained network grasped the underlying structure of images. This statement rests on the hypothesis that all images are in a low-dimensional manifold, and recent advances in realistic photos generation through generative adversarial networks bolsters this idea [cite Radford, Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks.].

More specifically, the Inception network learned to recognise features in a picture in order to classify the latter among the 1000 classes in the ImageNet dataset. Suppose that instead of classifying an image into 1000 classes we want to label it according to 6 different emotions (happy, sad, angry, scared, surprised, disgusted). The same features can be combined in a different way to let the network take a decision about what the emotion conveyed by the image is.

The process described above is called *Transfer Learning*: we chop off the last layer of the network and add our own layer given how many classes we have. We then freeze the weights of the other layers and only backpropagate through the newly created layer when training the network on our examples. If we have enough data, we can unfreeze more higher-level layers and backpropagate through them.

3.2 RESULTS

The Inception model was fed raw images, that were resized to a fixed size $(224, 224, 3)$, and fine-tuned with the following parameters:

- 9,000 training steps
- Mini-batch of size 32
- Adam optimizer with a learning rate of $1\mathrm{e}{-6}$

After the preprocessing described in Section 2, the dataset contains 295,508 posts that we split as 80% train set and 20% test set. The metric used to evaluate the model is accuracy, which is the fraction of correctly classified images.

The fine-tuned Inception is compared to a baseline: random guessing that includes the prior probabilities of the classes:

Table 1: Prior probabilities of the classes

	happiness	**sadness**	**anger**	**surprise**	**fear**	**disgust**
Prior proba.	0.32	0.22	0.19	0.03	0.22	0.02

The results are:

Table 2: Image model against random guessing

	Train accuracy	**Test accuracy**
Random guessing	24%	24%
Inception fine-tuned	48%	42%

4 NATURAL LANGUAGE PROCESSING

Even as a human being, it can be difficult to guess the expressed emotion only by looking at a Tumblr image without reading its caption as shown by Figure 2.

Figure 2: Which emotion is it?

It's unclear whether the user wants to convey happiness or surprise. Only after reading the text "To who ever left this on my windshield outside of last nights art opening, I love you. You made my night.", we can finally conclude that the person was *surprised*. The text is extremely informative and is usually crucial to accurately infer the emotional state.

4.1 WORD EMBEDDING

Most learning algorithms rely on the local smoothness hypothesis, that is, similar training instances are spatially close. This hypothesis clearly doesn't hold with words one-hot encoded as for instance 'dog' is as close to 'tree' as it is to 'cat'. Ideally, we would like to transform the word 'dog' in a space so that it's closer to 'cat' than it is to 'tree'. That is exactly how word embedding works: every word is projected into a high-dimensional space that preserves semantic relationships. Therefore, what the model has learned about dogs can be used when faced with a cat.

We will be using Word2Vec [elaborate]

Each post in the dataset does not necessarily contain the same number of words. Even after embedding each word, the input will be of variable size and most learning algorithm expect a fixed-sized input. To solve that problem, we can simply average across the number of words. The information loss is still minimal as the features come from a high-dimensional space [cite seth1]

However note that the word order is completely lost. Human language relies on the word order to communicate as for example the word *change* can be both a noun and a verb, and negation such as 'not entertained' can only be understood if 'not' directly precedes the verb. The order information can be preserved using Recurrent Neural Networks.

4.2 SEQUENCE INPUT

The text is broken down into a sequence of words that are embed in a high-dimensional space (unknown words are transformed to a zero vector) and then fed to an LSTM [cite]. To account for shorter posts, we zero-pad the vector with a special word token. For longer posts, we only keep the 50 first words [discuss, mean number of words in a post?].

We now evaluate the performance of the text model described below:

- Word embedding into a vector of dimension 300.
- An LSTM layer of size 1024.
- An output layer of size 6.

4.3 RESULTS

The network was trained with:

- 10,000 training steps
- Mini-batch of size 128

- Adam optimizer with an initial learning rate of 0.01
- Learning rate decay of $\frac{1}{2}$ every 1000 steps
- LSTM unrolled for 20 words
- Gradient clipping with a maximum norm of 5.0

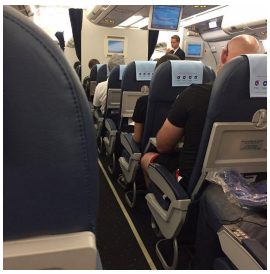
Table 3: Text model against random guessing

	Train accuracy	**Test accuracy**
Random guessing	24%	24%
LSTM model	63%	61%

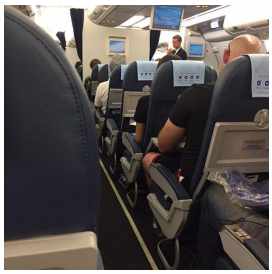
5 DEEP SENTIMENT

Real-world information oftentimes comes in several modalities. For instance, in speech recognition, humans integrate audio and visual information to understand speech, as was demonstrated by the McGurk effect [cite mcgurk]. Separating what we see from what we hear seems like an easy task, but in an experiment conducted by McGurk, the subjects who were listening to a */ba/* sound with a visual */ga/* actually reported they were hearing a */da/*. This is uncanny as even if we know the actual sound is a */ba/*, we cannot stop our brain from interpreting it as a */da/*.

Likewise, an image almost always comes with a caption as different interpretations can arise when textual context is not provided, as shown in Figure 3:



(a) "Planes might just be the most frightening thing ever." **scared**



(b) "I hate it when people are taking too much space on planes." **angry**

Figure 3: Different meanings with different captions.

Exploiting both visual and textual information is therefore key to understand the user's emotional state. Deep Sentiment is the name of the deep neural network incorporating visual recognition and text analysis.

5.1 ARCHITECTURE

Deep Sentiment builds on the models we have seen before as shown in Figure 4:

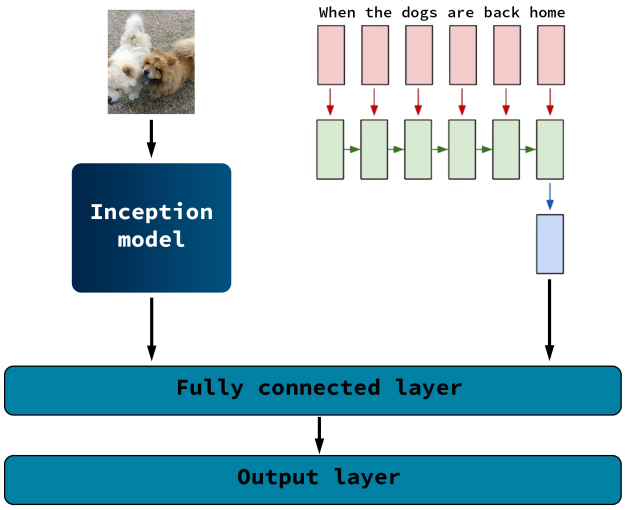


Figure 4: Deep Sentiment architecture

1. The image go through the pre-trained Inception model that extracts features from the images: more precisely with 256 neurons in the last Inception layer.
2. The text is embedded in a high-dimensional space with Word2Vec and is fed to an LSTM with 1024 neurons.
3. The two outputs are concatenated and fed to a fully connected layer with 512 neurons.
4. The final layer contains 6 neurons, one for each basic emotion.
5. Softmax is applied to the final layer to give the probability distribution of the emotional state of the user.

6 RESULTS

Deep Sentiment was trained with:

- 10,000 training steps
- Mini-batch size of 32
- Adam optimizer with an initial learning rate of 0.001
- Learning rate decay of $\frac{1}{2}$ every 1000 steps

This model combining text and image outperforms the algorithms only using those elements separately with 75% train accuracy and 70% test accuracy. This shows that just like a human being, a neural network needs both visual and textual information to determine the emotion conveyed by a post. The synergy between visual recognition and natural language processing is impressive as shown in the comparison table 4.

Table 4: Comparison of models

	Loss	**Train accuracy**	**Test accuracy**
Random guessing	-	24%	24%
Inception fine-tuned	1.61	48%	42%
LSTM model	1.01	63%	61%
Deep Sentiment	**0.80**	**75%**	**70%**

7 CITATIONS, FIGURES, TABLES, REFERENCES

These instructions apply to everyone, regardless of the formatter being used.

7.1 CITATIONS WITHIN THE TEXT

Citations within the text should be based on the `natbib` package and include the authors' last names and year (with the "et al." construct for more than two authors). When the authors or the publication are included in the sentence, the citation should not be in parenthesis (as in "See Hinton et al. (2006) for more information."). Otherwise, the citation should be in parenthesis (as in "Deep learning shows promise to make progress towards AI (Bengio & LeCun, 2007).").

The corresponding references are to be listed in alphabetical order of authors, in the REFERENCES section. As to the format of the references themselves, any style is acceptable as long as it is used consistently.

7.2 FOOTNOTES

Indicate footnotes with a number[1] in the text. Place the footnotes at the bottom of the page on which they appear. Precede the footnote with a horizontal rule of 2 inches (12 picas).[2]

7.3 FIGURES

All artwork must be neat, clean, and legible. Lines should be dark enough for purposes of reproduction; art work should not be hand-drawn. The figure number and caption always appear after the figure. Place one line space before the figure caption, and one line space after the figure. The figure caption is lower case (except for first word and proper nouns); figures are numbered consecutively.

Make sure the figure caption does not get separated from the figure. Leave sufficient space to avoid splitting the figure and figure caption.

You may use color figures. However, it is best for the figure captions and the paper body to make sense if the paper is printed either in black/white or in color.

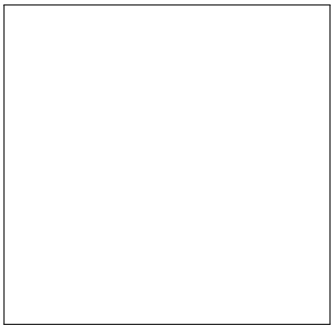


Figure 5: Sample figure caption.

7.4 TABLES

All tables must be centered, neat, clean and legible. Do not use hand-drawn tables. The table number and title always appear before the table. See Table 5.

Place one line space before the table title, one line space after the table title, and one line space after the table. The table title must be lower case (except for first word and proper nouns); tables are numbered consecutively.

[1] Sample of the first footnote

[2] Sample of the second footnote

Table 5: Sample table title

PART	DESCRIPTION
Dendrite	Input terminal
Axon	Output terminal
Soma	Cell body (contains cell nucleus)

ACKNOWLEDGMENTS

Use unnumbered third level headings for the acknowledgments. All acknowledgments, including those to funding agencies, go at the end of the paper.

REFERENCES

Yoshua Bengio and Yann LeCun. Scaling learning algorithms towards AI. In *Large Scale Kernel Machines*. MIT Press, 2007.

Johan Bollen, Huina Mao, and Xiao-Jun Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2:1–8, 2011.

Seth Flaxman and Karim Kassam. On #agony and #ecstasy: Potential and pitfalls of linguistic sentiment analysis. In preparation, 2016.

Geoffrey E. Hinton, Simon Osindero, and Yee Whye Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554, 2006.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *ICLR*, 2013.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.