OceanNet

Multimodal Emotion Recognition on Social Media with Deep Learning

George-Stefan Chiriluta
Supervisor: Angelo Cangelosi

April, 2020

Department of Computer Science
School of Engineering

A final project report submitted for the degree of
B.Sc. Computer Science

Acknowledgements

I would like to thank my supervisor Angelo Cangelosi for his amazing guidance, feedback and encouragements throughout my final year of studies and most importantly, during the development of this work. I want to express an enormous amount of gratitude to my parents, Simona and Vasea, and sister, Georgiana, for their never-ending support and motivation as well as to my best friends, Ahmad and Mihnea, for making these three years memorable.

Covid-19 Impact

The development of this project paper has not been severely affected by the Covid-19 pandemic as all work could be conducted remotely. However, the productivity diminished over time as the quarantine encouraged the authors to over-work and take fewer breaks.

Abstract

Massive amounts of multimodal information are available on social media platforms as people generally post images accompanied by textual descriptions and hashtags. The automatic extraction and analysis of affective knowledge from so much data can provide businesses, governments and individuals with insights into the market perception and audience feeling of their actions, products and services. Hence, we propose a Deep Learning approach to processing the visual and textual components of posts extracted from Tumblr in order to predict the emotions attached as hashtags by users. We perform transfer learning on the state-of-the-art InceptionV3 CNN for image classification to capture the highly abstract representations of emotions from pictures, while an LSTM network grasps complex temporal patterns of affects from the text sequences of these posts. Accuracy scores of up to 62.03% and 79.22% have been achieved on the problems of visual and textual emotion recognition, respectively, over the basic model of six emotions. Furthermore, we explore the fusion of these modalities in a multi-input network that combines the single-modality architectures and achieves 75.13% accuracy using this method. An in-depth discussion of the per-class performance of these models is provided as part of the evaluation. Finally, the research on these AI tools is integrated into the OceanNet framework. This incorporates a visualisation tool for market analytics platforms that produces real-time visualisations of affective statistics from social media information.

Contents

List of Figures

List of Tables

1 Introduction

This chapter highlights the recent history and evolution of Sentiment Analysis presented as a brief literature review. These background extracts introduce the motivation behind this project and the chosen Machine Learning algorithms and technology stack, which are followed by a brief overview of the work to be presented in more depth throughout this paper.

1.1 Sentiment analysis

The simplest form of sentiment analysis is classifying the polarity of documents. Pioneering studies in the field relied on building semantic orientation lexicons using semi-supervised approaches in order to predict the polarity of a piece of text. Such a partial knowledge-based method was formulated by Hatzivassiloglou and McKeown [1], who started from manually labelled sets of positive and negative seeds, and employed linguistic heuristics followed by clustering to generate lists of positive and negative adjectives. The semantic orientation of sentences would then be determined by the majority class, or by models that integrate the presence or frequency of these adjectives as features.

A trend of moving towards as little prior-knowledge as possible emerged around the beginning of the century, with Turney [2] proposing an unsupervised algorithm that computes the polarity of lexicon phrases as their point-wise mutual information with the words 'excellent' and 'poor'; the overall polarity of a phrase is then the average polarity of its lexicon phrases. One of the first prior-knowledge-free approaches was introduced by Pang, Lee, and Vaithyanathan [3], who trained three linear machine learning models (Naive Bayes, Logistic Regression and Support Vector Machine) on a dataset of **movie reviews** from which features such as unigrams, bigrams, part-of-speech tags and linguistic item locations were engineered. Their paper also brought to light some of the inherent difficulties of sentiment analysis, noting that the form of discourse must be taken into consideration, and, in agreement with Turney [2], that "the whole is not necessarily the sum of the parts".

The inability of these linear predictors with Bag-Of-N-Grams features to effectively integrate the word order [10] led to innovative solutions that leverage Convolutional Neural Networks (CNNs), which have been known to exceed human-level performance in many Computer Vision tasks. In a one-hot CNN [11], documents encoded as sequences of one-hot vectors are embedded into increasingly lower-dimensional representations as they are fed through the convolution and pooling pipeline. However, the regions processed by convolutional filters must be of fixed size and are also limited by the number of parameters to be learned. Flavours of recurrent neural networks, such as Long Short-Term Memory Neural Networks (LSTM), address these shortcomings, allowing for variable input sequence lengths on top of the main benefits provided by CNNs: they maintain information about the token order, and share parameters across the sequence so useful features can be detected invariant of their location.

The surge in popularity gained by social networks around 2010 led the natural language processing community to shift their attention towards Twitter, a microblogging service where people express their opinions on a variety of subjects - from the latest political controversy to the freshest Apple release. The large and diverse user base of people from different social and interest groups [6], and the mean frequency of about 6000 tweets per second make for retrieving training corpora of considerable sizes. Automatic collection of such datasets has been previously carried out by Read [4] from Usenet newsgroups, while Go and Huang [5], and Pak and Paroubek [6] extracted posts with the Twitter API, all of them using emoticons as

noisy labels; for example, :D is probably present in positive articles and tweets, while :/ in negative ones. However, outliers can arise and dealing with them is not trivial, for instance, when :) is used to express sarcasm instead of happiness. The 140 characters limit imposed on tweets means that they usually contain a single sentence, which has a high likelihood of correlation with the emoticon [6].

1.2 Multimodal emotion analysis

Although the state-of-the-art in sentiment analysis exceeds human-level performance, this binary classification system only captures how an entity is performing, rather than providing explicit, actionable insights. **This motivates the need to identify the nuances of feeling portrayed by human emotions.** Also, in human interactions, only part of the message is transmitted verbally as the vocal and **visual** elements often hold more weight [7]. Recent work conducted by Hu and Flaxman [8] considers the visual component encoded in the form of pictures from social media an essential source of information in predicting emotions which is complementary to textual descriptions. We follow their approach of automatically collecting a multimodal dataset using the Tumblr API, fetching those image posts accompanied by text that contain an emotion hashtag e.g. #amazed. Although such a dataset will contain some degree of noise due to variability in forms of expression - for instance, a writer could use #happy to express irony - generally, self-reported emotions represent the gold standard in psychology [8][9] and provide reliable labels for our corpus samples.

In this work, the relevance of the textual and visual modalities in expressing emotions is explored by training a set of Deep Learning models to perform single- and multi-modal emotion classification:

- For textual emotion analysis, we compose a network module comprising a stack of LSTM layers, on top of which a Softmax classifier renders a probability distribution over a discrete set of emotions.
- For visual emotion analysis, we perform transfer learning on a state-of-the-art CNN used for image recognition, namely InceptionV3, motivated by its ability to discover intricate structures in training data as our problem requires learning representations at a higher level of abstraction than image recognition. We replace the top layers with several Dense layers, with the last one computing a Softmax activation representing a probability distribution over the set of emotions.
- For multimodal emotion analysis, we concatenate the outputs of the single-modality networks with a Merge layer, on top of which a Softmax function produces a probability distribution over a discrete emotion spectrum.

1.3 OceanNet - An Emotion Analysis Tool for Market Analytics Platforms

There is a wide range of applications for these models, ranging from Business Intelligence solutions that can produce reports about the market perception to aiding mental health services to identify vulnerable people e.g. depression sufferers; furthermore, this research can serve a starting point in multimodal modelling of emotions in the Robotics and Human-Computer Interaction.

The research and software tools developed in this project were integrated into a novel framework, which we will call OceanNet. This frames the emotion analysis models within market analytics platforms by providing a tool that allows for an intuitive visualisation of computed emotion distribution statistics in the form of graphs and charts. The presented data is the result of feeding the single- and multi-modal networks with Tumblr data retrieved live, or uploaded by a salesperson. An online search functionality is provided to fetch posts containing a term of interest with a selected named-entity tag, using the Tumblr API. Named-entity recognition is performed on the retrieved posts by Spacy, an external service capable of detecting 18 types of entities, which has been trained on the large OntoNotes 5.0 corpus that contains various genres of text. Additionally, a market analyst can also upload a set of self-collected posts for analysis. The platform is implemented as a full-stack web application using ReactJS, Python Flusk and MongoDB; the architecture of the system and flow of communication are discussed in chapter 7.

2 State-of-the-art

This chapter presents the state-of-the-art works exploring the fusion of multiple modalities in sentiment and emotion analysis published before September 2019. Promising papers published or studied after the commencement of this project's implementation are covered as they represent potential improvements to the proposed systems.

2.1 Multimodal sentiment and emotion analysis on social media

The proliferation of social networks and its rich variation of multimedia content has opened new avenues for sentiment and emotion analysis as multiple modalities could be explored to augment the established text-based methods.

Pioneering work in multimodal sentiment analysis on social media data focused on the visual component, with Borth et al. [13] proposing the detection of adjective-noun pairs (ANPs) which describe the mid-level representations of images, such as salient objects, using a library of Support Vector Machine (SVM) and Logistic Regression classifiers [14]. The fusion of textual descriptions with these image-extracted ANPs has been shown to achieve better performance than text-only sentiment analysis on a dataset of tweets. This laid the foundation for MVSO [17], a Multilingual Visual Sentiment Ontology of ANPs containing a larger dataset of visual concepts along with a bank of Deep Learning ANP detectors.

In multimodal emotion analysis, Hu and Flaxman devised DeepSentiment [8], a neural network architecture that combines images and text in order to predict the emotion hashtags attached by bloggers to their Tumblr posts. This was integrated into a novel approach to exploring the structure of emotions on social media. The most relevant words per emotion category, the results of hierarchical clustering of emotions, and the projection onto the Circumplex Model [18] were proven consistent with what has been posited in the psychology literature and validated using the OASIS dataset [19]. The performance of DeepSentiment heavily influenced the architectural decisions behind the OceanNet models presented in this paper.

2.2 Multimodal sentiment and emotion analysis in the wild

Social media is a diverse source of multimedia content, predominantly popular for images and text; however, it is responsible only for a segment of the multimodal streams of data available

on the Web. Video sharing platforms are the preferred medium for video blogging and product reviews, which provide multimodal information of closer resemblance to human-machine and human-human interaction. MOSI (Multimodal Corpus of Sentiment Intensity) [15] introduced a dataset of 93 Youtube vlogs with per-segment subjectivity and sentiment annotations, per-frame and per-opinion visual features, and per-millisecond audio features, along with a collection of single- and multi-modal baselines. This line of research was expanded by the contextualised framework proposed by Poria et al. [16] which leverages the sequential order of utterances in the MOSI dataset. Context-independent unimodal features are extracted from the video transcript using a CNN fed with the concatenated word2vec embeddings of the individual words, while openSMILE performs audio feature extraction, and a 3D-CNN obtains visual cues from consecutive frames. These unimodal features of every utterance in a video are then fused and passed to the "Contextual LSTM" that captures the dependencies among input utterances. The concatenation of all modality channels (textual, vocal, and visual) achieves 80.3% accuracy with a bidirectional LSTM, exceeding that of single- and double-modalities.

The data streams used in these outlined works correspond to behavioural modalities that are, however, subject to multiple sources of bias. The content on social networks and video sharing platforms is mostly governed by how people want to present themselves rather than how they actually feel. Feelings are commonly suppressed willingly, and sometimes, simply by the nature of an individual. Liu et al. [21] emphasise that the complexity of emotions reaches beyond external appearances and emerges out of reactions from the central and peripheral nervous systems. Thus, they turn to datasets of physiological signals (SEED, DEAP, and DREAMER), which are more accurate and harder to be deliberately changed by subjects, and propose a framework of models that fuses modalities using Deep Canonical Correlation Analysis (DCCA) [20] [22]. This method has proven more resilient to noise in the training data, and its accuracy reaches almost 95% on the SEED dataset. Notable application areas of emotion recognition from physiological features include brain-computer interaction and the diagnosis of psychological disorders.

We conclude this subsection by noting that massive computational resources are required for compiling frameworks integrating physiological signals, and while they prove more precise, the behavioural modalities available on the Web represent a rich and easily accessible source of information that is more than sufficient for most multimodal sentiment and emotion analysis applications.

2.3 Promising works

Pre-trained word vectors such as Word2vec [23] and GloVe (Global Vectors for Word Embeddings) [24] have become an integrated component of modern natural language systems due to their ability to learn syntactic and semantic properties of lexical units from large scale unlabeled corpora [26]. While Word2Vec leverages the local context of words, Glove delves deeper and takes advantage of global word-word co-occurrence statistics; this family of word embeddings also overcomes the frailty of LSA-based (Latent Semantic Analysis) [25] methods in capturing semantic similarities between words. In the textual emotion analysis module of OceanNet, the tokens in the Tumblr descriptions are encoded using the 100-dimensional GloVe word vectors.

Although these static words vectors have pushed the performance of NLP systems in a variety of tasks, such as named-entity recognition and sentiment analysis, they provide a single representation per word, irrespective of context, and thus fail to model polysemy. General language models address this shortcoming of word embeddings by generating representations

that capture not only the identity of a word in a particular text, but also leverage its context in that setting [37]. ELMo (Embeddings from Language Modelling) [26] produces word representations that are functions of the entire input sequence which combine all layer activations of a biLM (bidirectional Language Model). The intermediate states computed by the first layer in ELMo's bidirectional LSTM has been shown to model aspects of syntax that are valuable in part-of-speech tagging or dependency parsing, while the second layer's output captures semantic information useful in word-sense disambiguation. However, ELMo is shallow bidirectional as the LSTM layers are trained left-to-right and right-to-left, sequentially and **independently** from each other. The state-of-the-art in language modelling, BERT (Bidirectional Encoder Representations from Transformers) [27] is deeply bidirectional as the input sequence is read all at once, and the attention mechanism behind Transformer jointly conditions on both left and right context in all layers. The novel performances achieved by BERT on eleven NLP tasks, including question answering and textual sentiment analysis, encourage its integration in the OceanNet framework in the place of static word embeddings, or as in the case of ELMo, alongside them. The paper for BERT has been published after the commencement of this project's implementation, but its adoption in the current model will be pursued in the event of a possible candidacy of OceanNet for a conference publication.

3 Emotion models

This chapter discusses the emotion representations adopted by contemporary Affective Computing systems, followed by the description and motivation behind the emotion representations used in the OceanNet models.

3.1 Categorical and dimensional models

In Affective Computing, emotions are usually represented using either the categorical or dimensional model. The former simply comprises discrete sets of emotions, while the latter places emotions in a multi-dimensional space.

The main benefit of the categorical model is the self-explanatory nature of the emotion classes. However, a comprehensive selection of emotion categories is crucial in producing consistent datasets and accurate classifiers. For example, devising a sparse set of emotion classes might not cover the entire spectrum of emotions expressed in a corpus, and can force annotators to wrongly label some samples; nevertheless, this can be solved by introducing a "neutral" or "unknown" class. On the other hand, a large number of emotion classes that partially overlap on the captured affects, or that are hardly-distinguishable from one another, can result in low inter-annotator agreement due to uncertainty and subjectivity in the manual labelling process, or in the inability of a statistical model to learn separable properties of the emotion categories. In spite of these disadvantages, the categorical model has been prevalent in the field [28]; the emotion words are generally familiar to the large audience, and their ubiquity as hashtags on social media enables the automatic collection of large labelled corpora that regard the self-reported emotion tags as the ground truth. This suggests that categorical models are the suitable candidate for our automatic dataset collection process and emotion analysis problem.

The dimensional model captures a wide range of fine-grained emotions that would be difficult to represent within the exact boundaries of the categorical model. In the dimensional model, each emotion class corresponds to a region of the defined space, with each point in these regions introducing variation according to the polarity, intensity, and other properties of the perceived affect. This circumvents the uncertainty issue that arises when the choice of an

annotator is in-between two emotion classes since an intermediate value can be selected. The dimensions used in this model usually correspond to valence (or polarity), arousal (or level of excitement) and power (denotes the degree of control over the emotion) [28]. Also, the ability of the dimensional model to compute similarities and differences between emotions represents another notable feature missing from the categorical model. However, training human annotators to label corpora based on multiple dimensions may be difficult, and to cope with potential sources of bias and subjectivity, an average of several measurements coming from different annotators is needed. Furthermore, the dimensional model is not directly applicable to the automatic extraction of social media datasets.

In the following subsections, we cover the two categorical models on which the OceanNet models have been trained.

3.2 The basic emotion model

The revolutionary study of Paul Ekman [29] regarding the universality of facial displays of emotion across cultures has profoundly motivated psychology research to focus on the similarities of human processes as well rather than solely on the differences [30]. The agreement among subject groups of literate and preliterate background has confirmed that emotions are biologically hard-wired in humans. The list of basic emotions used in this study, also known as "The Big Six", comprises: "happiness", "fear", "disgust", "anger", "surprise", and "sadness".

The pan-cultural universality of these emotions led to their adoption as the basic emotion model used in many Affective Computing systems, with Strapparava and Mihalcea [31] pioneering its integration in knowledge-based and statistical methods of emotion recognition. As these emotions cover a generic and comprehensive spectrum of affects, they are transferable across the majority of application fields of emotion mining. Since marketing often spans a variety of activity domains, this emotion model is suitable for our market analysis platform, OceanNet. Furthermore, additional emotion classes required by more specialised fields - for example, "boredom" is an important measurement of engagement in e-learning resources - can be easily integrated into our framework: posts containing the respective emotion hashtag can be retrieved to expand the dataset, while the topology of the networks can easily incorporate output neurons corresponding to the new classes' probabilities in the final prediction layers. Finally, the frequency of these emotions on social networks facilitates the automatic collection of a large corpus; the abundant amount of samples per class promises a robust and balanced dataset to be be used in training our single- and multi-modal models.

3.3 DeepSentiment's emotion model

DeepSentiment [8] proposes a more comprehensive emotion model in order to explore the structure of emotions on social media. The list comprises 15 emotions selected from the PANAS-X scale [32] and Plutchik's Wheel of Emotions [33] with high frequencies on Tumblr. These are: "happy", "calm", "sad", "scared", "bored", "angry", "annoyed", "love", "excited", "surprised", "optimistic", "amazed", "ashamed", "disgusted", and "pensive". The dataset collected for the basic emotion model has been enhanced with the additional classes in DeepSentiment's model. Training the single- and multi-modal networks on this augmented dataset provide a frame of comparison with DeepSentiment's performance, thus allowing us to validate the hyperparameters and architectural decisions behind OceanNet.

4 Datasets

This chapter describes the automatic collection, sanitation and preprocessing steps involved in building two multimodal datasets from Twitter and Tumblr, and elaborates on the appositeness of data from these social networks for multimodal emotion analysis. Table 1 below summarises the final distribution of emotions in the Tumblr and Twitter datasets at each collection iteration.

	Tumblr		Twitter		
	v1	**v2**	v1	v2	v3
Happy	20007	**40003**	3368	4321	15863
Calm	14324	**14325**	190	241	250
Sad	20001	**40003**	687	799	834
Scared	8156	**8156**	46	53	588
Bored	20001	**39692**	314	354	414
Angry	11843	**11844**	154	175	233
Annoyed	3965	**3965**	26	30	57
Love	20005	**40005**	14156	16528	17016
Excited	19163	**19163**	971	1039	1130
Surprised	2579	**2579**	8	10	240
Optimistic	1868	**1868**	31	37	1902
Amazed	749	**749**	8	16	1886
Ashamed	477	**477**	12	12	12
Disgusted	670	**670**	24	29	31
Pensive	1106	**1106**	0	0	2
Size:	144914	**224605**	19995	23644	40458

Table 1: Distribution of emotions in the Tumblr and Twitter datasets

4.1 Tumblr

Tumblr is a blogging platform with an abundance of expressive textual content that often comes accompanied by pictures and hashtags. The length of the descriptions is not limited, making Tumblr posts more suitable than their shallower Twitter counterparts [8]. Moreover, a strong correlation between text and images is established in many posts; for instance, in figure 2, analysing the text alone suggests happiness and surprise, while the red and dark tone of the picture might imply anger or sadness. In this example, the image emphasizes the feeling derived from the description.

4.1.1 Automatic collection

To collect the first version of our Tumblr dataset, an extraction script makes queries to the Tumblr API requesting posts that contain the emotion tags from the two emotion models presented in chapter 3. We keep only those posts that consist of both pictures and textual descriptions, beginning with December 2019 and going back as far as January 2011. Further, non-English posts - the ones with less than 50% English words - are filtered out. Glove's

Figure 1: Tumblr post consisting of picture, textual description and hashtags. Source: lapfullofcatfur [45]

Figure 2: Post illustrating the strong correlation between picture and description. Source: girlwithlandscape [46]

vocabulary [24] was used for this task, and even though it contains some non-English words, very few non-English posts made it through. The correctness of this approach has also been confirmed by Hu and Flaxman [8]. To augment this corpus, more requests are sent for posts with hashtags of synonyms of the sought emotions, materialising in version two of our dataset. The size of the dataset jumped from 144,914 to 224,605 (depicted in table 1) as the result of this procedure. The list of synonyms is included as an appendix. Finally, while the emotions corresponding to the basic emotion model are well-represented (except "disgusted"), those more specialised belonging to DeepSentiment's model are under-represented, for instance, "amazed" and "ashamed" are associated to less than 1000 samples each. This motivates the need for handling this imbalance during training; thus, in chapter 6 (Results & Evaluation) we explore the benefits of weighted loss functions and resampling techniques.

4.1.2 Text preprocessing

The text descriptions undergo several transformations before being mapped into word embeddings:

1. Special strings, such as URLs and emoticons, are discarded.
2. Punctuation is removed in order to be left only with content and function words.
3. Words corresponding to the emotion tag that appear in the text are removed such that we avoid an unfair influence over the performance of our models [8].
4. The text is tokenised into lexical units.

4.2 Twitter

The reasons behind the popularity of Twitter for sentiment analysis have been introduced at the end of section 1.1. The massive amount of content made available by people who share opinions on a variety of subjects - from a new purchase review to speculations about a world pandemic - has continuously been leveraged by companies, governments and individuals in order to understand their audience. However, there are several downsides with respect to its suitability for multimodal emotion analysis:

- Tweets are limited to 140 characters, which limits expressivity.
- Although tweets that contain multimedia content such as gifs, images and videos enjoy higher retweet rates [38] and more popularity, more than 90% of tweets are text-only [39].
- A tweetstorm is a series of connected tweets posted by a user in a short interval of time to get around the 140 character limit. Thus, the tweets of tweetstorms could be aggregated into more meaningful samples that present suspenseful stories or discuss rampant issues. However, the tweets of tweetstorms are generally single-modal as they only contain text in most cases, making them unsuitable for our multimodal task.

In spite of these downsides, we still collected a multimodal dataset from Twitter to analyse the impact of these considerations and whether the affective knowledge learned by our models is transferable among social networks.

Figure 3: Twitter post for emotion “calm”. Source: YorkshireShepherdess [47]

Figure 4: Twitter post for emotion “ashamed”. Source: Kevin Nevis [48]

4.2.1 Automatic collection

The collection process of the Twitter dataset is similar to Tumblr: an extraction script sends requests to the Twitter API for posts that contain an emotion tag, with only those in English which consist of both images and text being kept. The standard version of the API can only go back as far as 7 days, thus placing a limit on the size of our corpus. We collected three incremental versions of our dataset (statistics are presented in table 1):

1. Version 1 contains 19,995 posts that are highly unbalanced between the 15 emotion classes. This has been used as a pilot for preprocessing.

2. Version 2 has been augmented with more posts as they were made available by the Standard Twitter API, increasing the size of the dataset to 23,644. However, the distribution of the posts is still disproportionate for half of the classes.

3. The final version has been augmented with posts that contained synonyms of the emotion classes, pushing our dataset size to 40,458 posts. Although the new distribution of samples is also uneven, we have significantly more samples for most of the previously under-represented emotions that can serve as a starting point for further augmentation techniques. However, there are still four barely-present classes: "annoyed", "ashamed", "disgusted", and "pensive", which our classifier will fail to learn. Weighted loss functions and resampling methods are explored to circumvent this issue.

4.2.2 Text preprocessing

The Twitter posts are preprocessed in a similar fashion to their Tumblr counterparts, with the single addition that special strings to be removed include "RT" (standing for retweet), and user tags such as "@realDonaldTrump", on top of URLs and emoticons. The rest of the steps are unchanged, with punctuation and emotion tags being discarded from the text, which is ultimately tokenised into lexical units.

5 The OceanNet Models

This chapter presents the Deep Learning models addressing the problems of textual, visual and multimodal emotion analysis. For textual emotion recognition, we propose a network integrating word embeddings with Long Short-Term Memory layers, while for the visual task we perform transfer learning on a state-of-the-art Convolutional Neural Network for image classification, namely InceptionV3 [36]. The network for multimodal classification combines these single-modal architectures by concatenating their final activations in a single tensor which is used in making the prediction.

5.1 Textual emotion analysis

Research in sentiment and emotion analysis has mainly focused on the textual descriptions of social media posts as they represent an abundant, diverse and expressive source of affective information. Although these subjective texts often provide sufficient cues to clearly communicate the emotional state of the user at the moment of posting, sometimes affects are hidden in sophisticated formulations. For instance, "I'm bored out of my mind and can't stop thinking about anime." explicitly states the feeling of boredom, while "In Season 2 of my slow-brew Buffy rewatch, and **stumbled across this most perfect of moments.**" (figure 2) subtly

conveys a positive surprise through the second clause. This second example exposes the inherent difficulty of recognising emotions from natural language, as single words or groups of words can carry the entire emotional weight of the message or point towards a range of emotions. Thus, the syntactical and semantic relations between words must be captured, which motivates the use of word embeddings, as well as the weight they carry in conveying specific emotions. Also, information about the dependencies among such words in their occurring contexts must be maintained, and this has been achieved recently in related works by the use of recurrent neural networks [34]. Hence, we propose a deep architecture combining a look-up table of word embeddings followed by a stack of LSTM layers in order to predict the emotions expressed by text posts.

5.1.1 Word Embeddings

Static word embeddings capture the syntactic and semantic relationships between words by mapping them into a highly-dimensional vectorial space where similar words are located in each other's proximity, while distant words are situated widely apart; for instance, the word "cat" must be closer to "dog" than to "boat". Hu and Flaxman [8] indicated that this property of word embeddings is desirable as "most learning algorithms rely on the local smoothness hypothesis, that is, similar training instances are spatially close", which cannot be achieved by simply one-hot encoding the lexical units. Thus, the performance boost fueled by pre-trained word embeddings contributed to their integration in most state-of-the-art natural language systems.

For these reasons, we will initialise the weights of our word embedding layer with GloVe's [24] 100-dimensional word vectors. GloVe (Global Vectors for Word Representation) has been trained on a large corpus of tweets, hence, the transfer of the relations already captured by these vectors proves useful when training our model on the Twitter dataset. This applies for the Tumblr corpus as well since the overall nature of the content on the two platforms is similar, with DeepSentiment supporting this assumption. During training, the weights of the embedding layer are unfrozen in order to fine-tune them for the task of multimodal emotion analysis, and once optimised, they can be visualised in the Google Embedding Projector.

The Google Embedding Projector performs Principal Component Analysis on the supplied word vectors and renders the first three components. The figures 5, 6, and 7 below depict the visualisation of a set of word embeddings learned from scratch from the IMDB movie review dataset as a learning experiment for the task of sentiment analysis. The words "director" and "actor" have been positioned close together in figures 5 and 6 because they belong to the same field, while "cook" is located on the opposite pole as it pertains to a different domain. The data points have been mapped to the surface of a sphere for this distinction to be evident, with the two poles clustering words of the same polarity or that are semantically related. In chapter 6 (Result & Evaluation), we will visualise the fine-tuned GloVe word vectors for the Tumblr and Twitter datasets.

5.1.2 Implementation considerations for word embeddings

The Keras word embeddings layer is an abstraction over a look-up table that maps integer encoded words into their corresponding word vectors; thus, the shape of this layer is determined by the vocabulary size and the dimensionality of the word embeddings. As the Twitter dataset comes with a limited vocabulary that consists of only 27582 words, using the 100-dimensional GloVe word vectors requires a look-up table of 27,584 by 100 (two embeddings have been added

to account for the "unknown" and "padding" tokens). The number of weights to be fine-tuned in this case is a reasonable 2,758,200. On the other hand, the vocabulary of the Tumblr corpus is richer as it comprises 225,542 words, expanding the shape of the embeddings layer to 225,544 by 100 when using the same word vectors, and lifting the number of weights to be optimised to a major 22,554,400. This number still fits in the memory of modern computers, however, it demonstrates that a large vocabulary size translates into a considerable memory requirement as these 22,554,400 weights consumed approximately 1.6 GiB of RAM space.

Although recurrent neural networks have the ability to accept variable-sized sequences, the Keras implementation of the word embeddings layer that precedes our LSTM stack requires the input length to be specified in advance. We follow the decision of DeepSentiment of capping the textual descriptions to 50 words since 91.72% of our Tumblr texts and all of our tweets are of this length or shorter. Padding is applied to posts comprising less than 50 words.

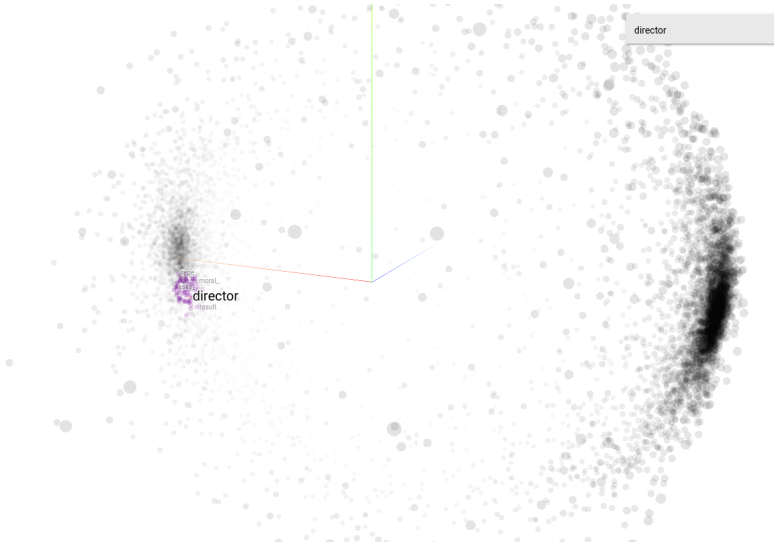

Figure 5: Position of the word "director" in a space of word embeddings learned from the IMDB movie review dataset. Visualisation created using the Google Embedding Projector.

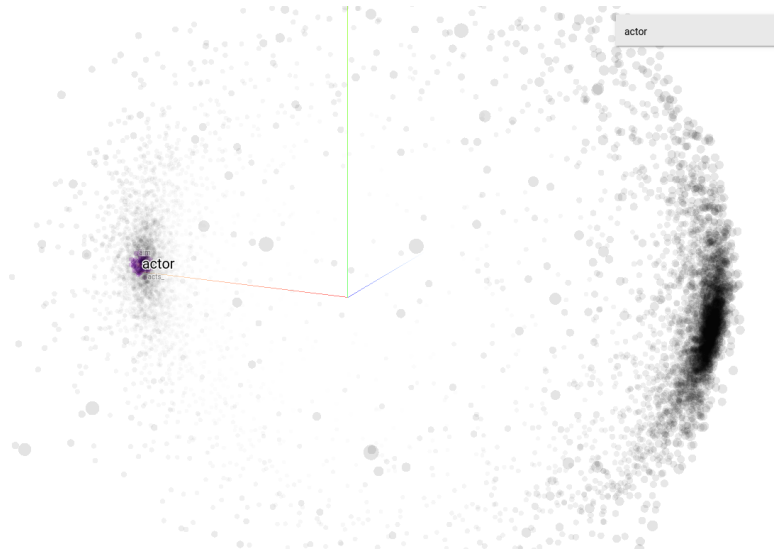

Figure 6:]
Position of the word "actor" in a space of word embeddings learned from the IMDB movie review dataset. Visualisation created using the Google Embedding Projector.

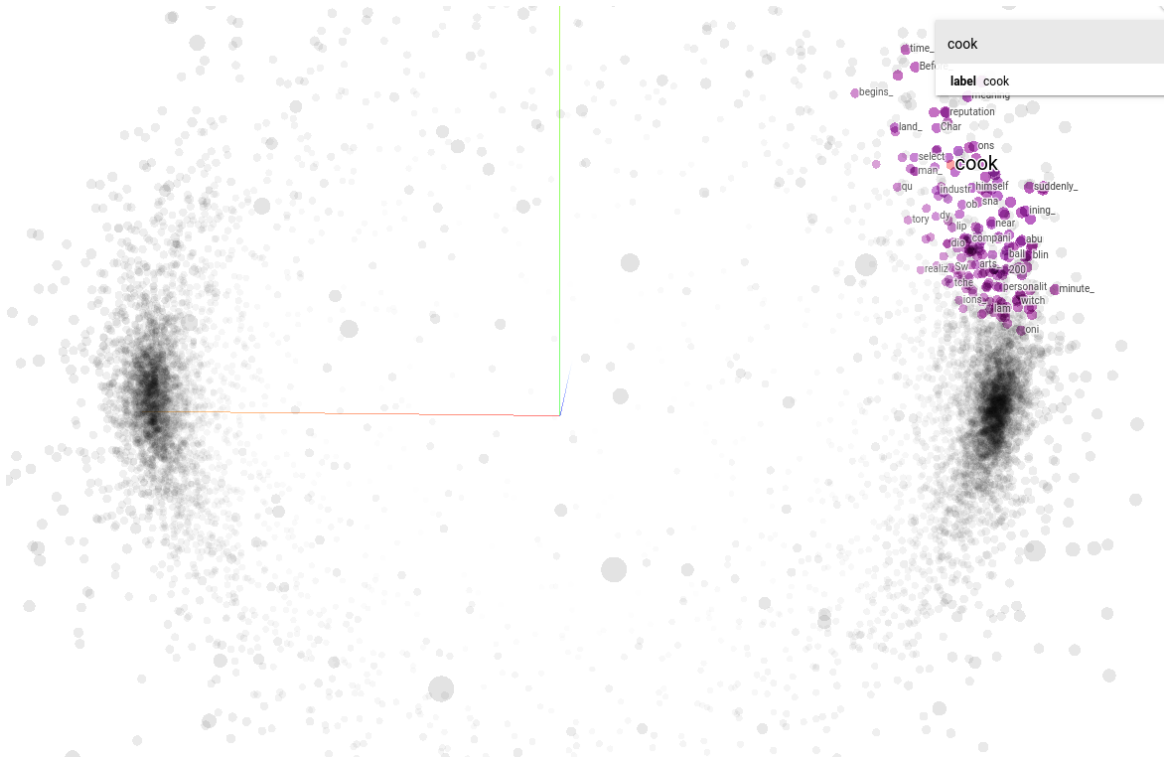

Figure 7: Position of the word "cook" in a space of word embeddings learned from the IMDB movie review dataset. Visualisation created using the Google Embedding Projector.

5.1.3 Long Short-Term Memory Stack

Recurrent Neural Networks address the shortcomings of traditional statistical approaches in modelling signals that are sequential in nature. While vanilla neural networks take in fixed-size vectors and learn different connections for different parts of the input, RNNs are able to model variable length data and share parameters across the input sequence in order to capture important features invariant to their location. The internal hidden state maintained by RNNs stores information about the previous context as the units of the sequential input are fed into the network in order. This bears a close resemblance to the short-term memory employed by humans when reading as past keywords and ideas are essential in understanding the current context. Since the textual descriptions in our Tumblr and Twitter datasets represent such sequential signals, we will leverage a flavour of RNN able to model long-term dependencies useful in determining the underlying emotion.

Vanilla RNNs suffer from the vanishing gradient problem as for long sequences the early layers will only receive small gradient updates via backpropagation-through-time (BTT), hence learning very slowly and ultimately stop acquiring new knowledge altogether. Thus, RNNs forget information distant in time and are viewed to have short-term memory. The Gated Recurrent Units (GRUs) and Long Short-Term Memory Neural Networks (LSTM) are two flavours of RNN that overcome this issue using an internal mechanism based on control gates (figure 8) that regulate the flow of information in the recurrent layers [41], and are thus able to track long-term relationships among the input sequences. These topologies maintain a cell state that is updated with new data processed by a series of gates (functions) that decide which data is relevant and must be retained, and which is not and can be discarded.

Long Short-Term Memory layers produce their output and update their internal state using a pipeline of three gated operations. First, irrelevant data from the previous state and current

input is discarded using a sigmoid function that shrinks its input in the (0, 1) interval. Secondly, the important information is retained and used to update the memory cell state. Finally, this new cell state is used along the previous hidden state and current input to produce the new hidden state. This series of operations is depicted in figure 8 below. We explore a set of architectures comprising stacks of LSTM layers in order to determine the most accurate topology for identifying the expressed affects, and report the results in chapter 6 (Results & Evaluation). According to empirical evidence [35], a deeper stack should translate into increased performance. However, this is invalidated by our finding that a single layer of 1024 units achieves better performance than stacks of 3 and 5 LSTM layers with 124 or 1024 units each.

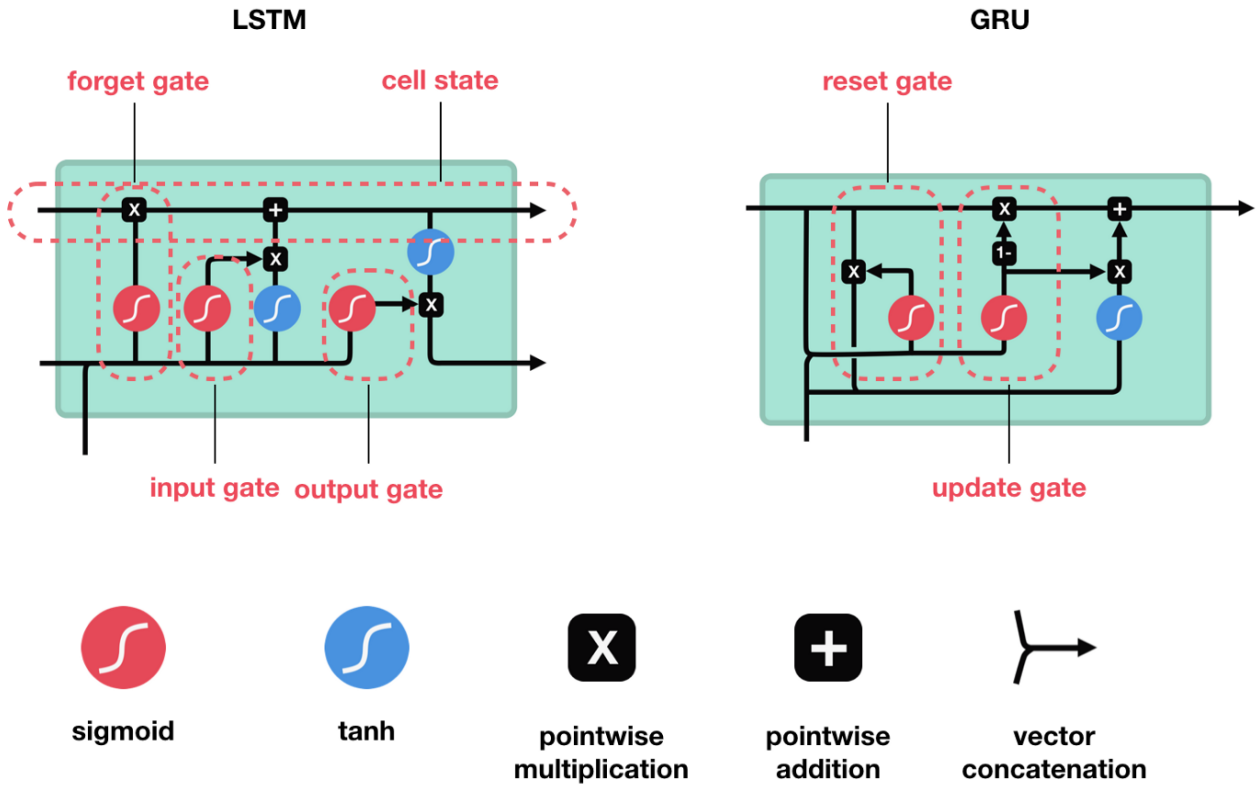

Figure 8: LSTM and GRU memory cells. Source: Medium [41]

5.1.4 Overall architecture

The network architecture that we propose as a component of the OceanNet framework for the task of textual emotion recognition comprises an initial look-up table of word embeddings, which projects the words of post descriptions onto a high-dimensional space satisfying the local smoothness hypothesis, followed by a single LSTM layer of 1024 units that capture complex temporal representations from its sequential input of word vectors. A dense layer consisting of 1024 neurons receives the activation of the top LSTM module, and a final Softmax produces a probability distribution over the selected emotion model. This topology is presented in figure 9 which depicts an unfolded LSTM module that is fed with word embeddings corresponding to the input tokens, and which transmits its output activations to the final classifier. The rest of the explored architectures for the LSTM stack are discussed in chapter 6.

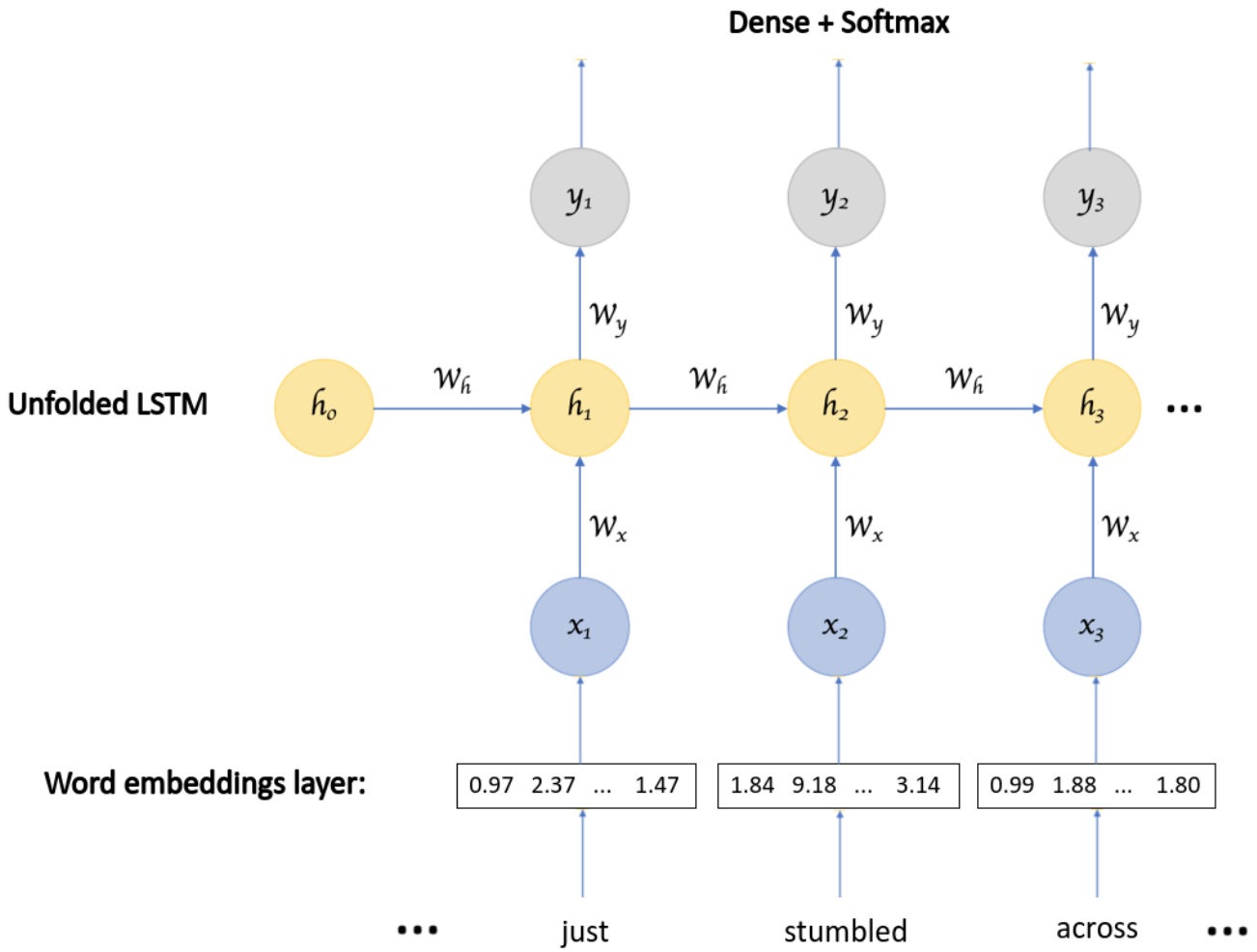

Figure 9: Overall architecture of the model for textual emotion recognition. Words in the input sequence are projected onto word embeddings, which are then fed in the following LSTM layers. A dense layer with Softmax activation renders a probability distribution over the desired emotion model. The unfolded LSTM in the picture has been extracted from [40].

5.2 Visual emotion analysis

As pictures on social media commonly emphasise the affect communicated by the textual description, identifying the underlying emotion solely from the image represents a challenging task even for humans; for example, in figure 4, it is impossible to infer that the emotional state expressed by the author is "ashamed" without the text of the tweet. This suggests that models attempting to recognise feelings from images alone must capture representations of intricate structures at a high level of abstraction [8]. The inherent difficulty of the problem is explored by performing transfer learning on a state-of-the-art CNN for image classification, namely InceptionV3 [36], in order to assess the ability of visual stimuli to convey emotions by themselves.

InceptionV3 is the top runner of the ILSVRC (ImageNet Large Scale Visual Recognition Competition) 2015, which has been trained on a subset of the ImageNet dataset containing roughly 1.2 million training images corresponding to 1000 classes. The main innovation behind this model is based on the discovery that the salient parts of an image vary in size; hence, the network comprises multiple "Inception modules", each of which performs convolutions with multiple kernel sizes in order to capture globally and locally distributed information. Also, 1x1 convolutions were put in place in order to reduce the depth dimensionality of the filter banks; thus, fewer weights have to be optimised and the training time is shortened. The architecture

of Inception is depicted in figure 10. Input images have to be resized to 299 by 299 pixels on three channels prior to being passed to the network.

The low- and mid-level patterns captured by the shallow and intermediate layers of Inception represent edges, motifs, and parts assembling most structures present in images. The universality of these representations in the visual stimuli motivates the transfer of this knowledge to our emotion recognition problem. The first 9 "Inception modules" are responsible for capturing these general features, hence, we freeze their corresponding layers during training. On the other hand, the top two modules must learn high-level aspects of our Tumblr and Twitter images, so they are left unfrozen. The top classifier used in ILSVRC is replaced by a Dense layer followed by a Softmax rendering a probability distribution over the selected emotion model. The results of this process are reported in chapter 6 (Results & Evaluation) on both the basic and DeepSentiment's emotion models.

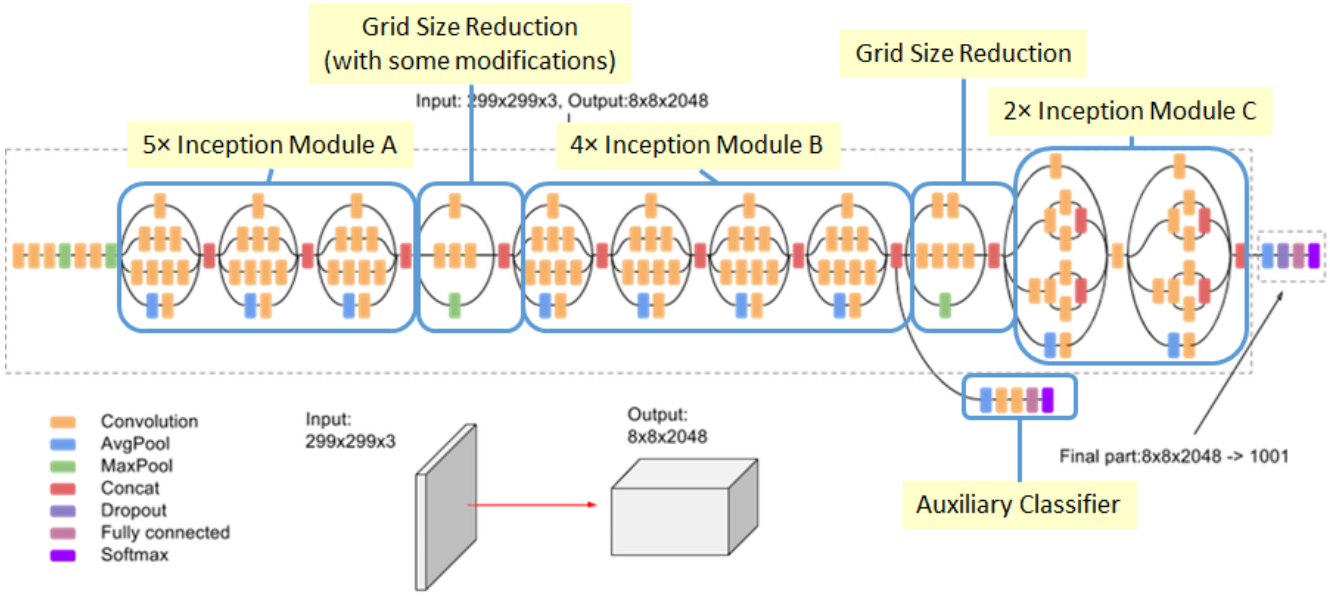

Figure 10: InceptionV3 architecture. The first 9 blocks capture low- and mid-level patterns that are universal across many computer vision tasks, while the top 2 blocks are fine-tuned for the problem of visual emotion recognition. Source: Medium [42].

5.3 Multimodal emotion analysis

Human to human interaction is the most natural example of information transmitted using multiple modalities as it comprises both visual and verbal signals. On social media, the picture often emphasises or complements the textual descriptions, which should increase the accuracy and confidence of the predictions. Thus, we propose a multi-input model that predicts emotions from both images and text by combining the single-modal networks proposed for textual and visual emotion recognition. The top classifiers of these networks are discarded, and the activations of the remaining final layers are concatenated and merged into a fully-connected layer of 1024 units, followed by a Softmax that produces a probability distribution over the selected emotion model. This architecture is similar to the topology of DeepSentiment (figure 11), with the exception that the CNN processing the visual component is InceptionV3 rather than InceptionV1.

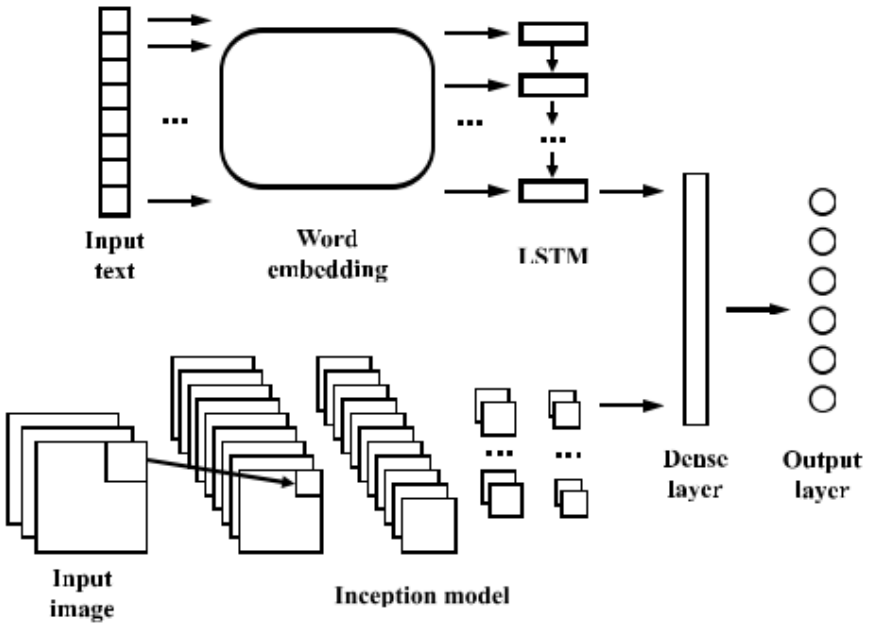

Figure 11: DeepSentiment's architecture. The topology of the OceanNet multimodal network is similar, except for InceptionV3 being used instead of InceptionV1. Source: Hu et al. [8].

6 Results & Evaluation

This chapter describes the performance achieved by the textual, visual and multimodal networks presented in the previous section on the Tumblr and Twitter datasets. The results produced after training these models on the basic and DeepSentiment's emotion models are showcased and interpreted using a range of evaluation metrics such as accuracy, precision, and recall scores. Hyperparameter exploration and class balancing experiments are also covered.

6.1 Textual Emotion Analysis

Training the model for textual emotion analysis on both the Tumblr and Twitter corpora provides a frame of comparison between the emotional expressivity of the descriptions posted on the two social networks. This allows us to validate the claims made in chapter 4 regarding the appositeness of these platforms for the task at hand. Also, testing the network trained on the Tumblr dataset using its Twitter counterpart and vice-versa provides us with an assessment mechanism for the transferability of the learned knowledge across the two platforms.

6.1.1 Hyperparameter grid search for the LSTM stack

This section presents the experiment setting of the hyperparameter grid search for finding the top performing architecture of the textual emotion analysis model and discusses the obtained results.

6.1.1.1 Experiment set-up

The model proposed for textual emotion analysis comprises a look-up table of word embeddings followed by a series of LSTM layers. As empirical evidence suggests that deeper recurrent topologies achieve higher accuracies than shallower ones [35], we perform a grid search in order

to determine the top-performing architecture for our LSTM stack. The hyperparameters that need to be optimised are the number of layers along with the count of neurons they should each contain. The values considered for the number of layers are 1, 3, and 5, with either 124 or 1024 units each. For this experiment, we use the Tumblr dataset due to its larger size and because its distribution of samples across the emotion classes is more balanced than in its Twitter counterpart. Also, DeepSentiment's emotion model is used in this classification as it covers a broader spectrum of affects which requires the network to grasp more intricate relationships.

To prevent the network from overfitting we introduce a simple but effective form of **regularisation** called Dropout, which simply turns off part of the neuron activations in order to force the network not to rely on any specific nodes. We introduce 20% dropout in all recurrent and forward connections of the LSTM layers. Furthermore, the training process is halted if the validation loss increases continuously for 3 epochs while the training loss keeps decreasing. In this way, we detect when the network starts to learn the training set too well i.e. memorise it since it does not generalise to the previously unseen data of the validation set. This technique is known as early stopping, and helps in avoiding overfitting too.

6.1.1.2 Grid search results

Table 2 presents the results of this hyperparameter grid search and suggests that a single layer with 1024 neurons represents the optimal choice as it achieves the highest validation accuracy of 68.56%. This contradicts the empirical evidence according to which stacking recurrent layers translates into better performance [35]; in our experiments, going deeper constantly reduces the accuracy scores of the model. Also, the discrepancy between the accuracies obtained for training and validation data indicates a slight trace of overfitting. In subsection 6.1.2 (Regularisation), we explore multiple dropout rates in an attempt to give our model a stronger generalisation ability.

Thus, the final architecture of the OceanNet module responsible for textual emotion analysis consists of a look-up table of word embeddings, a single LSTM layer with 1024 units, and the fully-connected layer adapted to the desired emotion models with a softmax activation. This is the topology used in the following experiments. Training was performed on Kaggle's Intel Xeon CPUs as parallelisation is limited by the sequential nature of processing in LSTMs.

	1 layer			**3 layers**			**5 layers**		
Neurons	Training	Val	Time	Training	Val	Time	Training	Val	Time
128 units	76.03%	66.69%	27m	74.01%	65.73%	1h 12m	74.84%	64.52%	2h 25m
1024 units	78.83%	**68.56**%	32m	78.76%	66.74%	1h 48m	68.13%	58.74%	6h 7m

Table 2: Accuracies and training times obtained by the hyperparameter grid search for the top performing architecture of the LSTM stack. The hyperparameters to be optimised are the number of LSTM layers and the count of neurons per layer.

6.1.2 Regularisation

Regularisation techniques are used to prevent or reduce the degree of overfitting acquired by a statistical model. Dropout is one of the simplest yet most effective forms of regularisation, hence, we explore the extent to which we must apply it in order to improve the generalisation ability of our model. Table 3 below presents the training and validation accuracies obtained

using a range of dropout rates in the recurrent and forward connections of our LSTM layer. While the discrepancy between the training and validation accuracy decreases as we increase the dropout percentage, so does the performance of the model on the validation set. Thus, we can conclude that excessive regularisation does not translate into better results. As the best accuracy is achieved at 20% dropout, this is the rate used when training the OceanNet network for textual emotion recognition in the next experiments to be presented.

	Dropout rate			
	0.2	0.3	0.4	0.5
Training accuracy	**78.83**%	76.81%	76.43%	73.32%
Validation accuracy	**68.56**%	68.33%	67.98%	67.20%

Table 3: Training and validation accuracies obtained by a single LSTM layer of 1024 units over a range of dropout rates.

6.1.3 Handling the imbalanced dataset distribution

The additional affects encompassed by DeepSentiment's emotion model are under-represented in the collected corpora in comparison with the basic set of emotions. The "amazed", "ashamed", "optimistic" and "pensive" classes contain less than 2000 samples each, while "happy" and "sad" are very popular with over 40000 posts in the Tumblr dataset. Although the infrequency of an emotion on social media should translate into the smaller probability of any given post to express that affect being captured by the network, we address this imbalance across the distribution of classes in the corpus as otherwise, the model may learn to predict only a subset of the affects or perform better for some emotions than the others. Thus, we explore the use of weighted loss functions as well as two over-sampling technique, namely random and SMOTE, in order to balance the data in our corpora. We note any performance improvements achieved by these methods over standard training without class balancing. The Tumblr corpus is used for these experiments since it contains more posts for the under-represented classes than its Twitter counterpart, and the final layer of the network is adapted to DeepSentiment's emotion model.

6.1.3.1 Performance without class balancing

The hyperparameter exploration discussed in the previous sections has been performed without class balancing the data in the Tumblr corpus. The top accuracy was achieved by a single LSTM layer with 20% dropout and stands at 68.56%, which is in line with DeepSentiment's score of 69%. This represents a good result for a 15-class problem that requires the underlying model to capture highly abstract representations. However, we also need to assess the classifier's ability to detect each emotion class as we are equally interested in all affects. Thus, per-class metrics have been computed and displayed in table 4, while the confusion matrix of this classification on the validation set can be visualised in figure 13.

Emotion classes that are frequent in the Tumblr dataset achieve both high recall and precision, while the under-represented affects achieve an even higher recall but very poor precision. This trend is also outlined by the F-1 score (the harmonic mean of precision and recall) which increases with the number of class samples. These relationships are illustrated in figure 12 below and interpreted on the next page.

Emotion class	Samples	Recall	Precision	F1-score
Happy	40,003	0.74	0.71	0.72
Calm	14,325	0.71	0.58	0.64
Sad	40,003	0.59	0.76	0.67
Scared	8,156	0.63	0.39	0.49
Bored	39,692	0.65	0.71	0.68
Angry	11,844	0.70	0.40	0.51
Annoyed	3,965	0.67	0.37	0.48
Love	40,005	0.77	0.76	0.76
Excited	19,163	0.60	0.75	0.67
Surprised	2,579	0.78	0.33	0.46
Optimistic	1,868	0.84	0.43	0.57
Amazed	749	1.0	0.04	0.08
Ashamed	477	1.0	0.08	0.14
Disgusted	670	1.0	0.27	0.42
Pensive	1,106	0.92	0.27	0.41

Table 4: Precision, recall and F1-score metrics achieved by a single LSTM layer of 1024 units trained **without class balancing** on the Tumblr dataset.

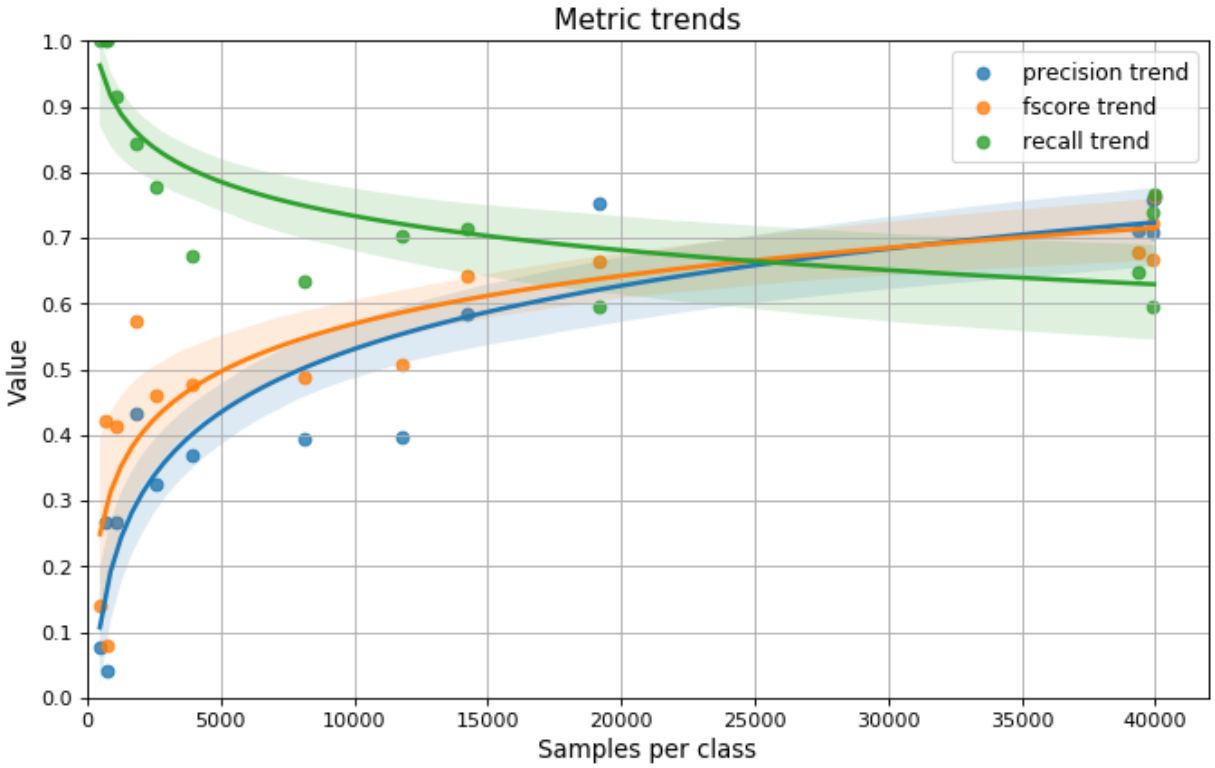

Figure 12: Precision, recall and F1 trends over the number of samples per emotion class.

The good precision and recall scores achieved by popular affects such as “happy” and “love” indicate that the model has learned to correctly identify the frequent classes. On the other hand, under-represented emotions suffer from low precision as the model often mistakes them for the more popular categories. This can be observed in the confusion matrix depicted in figure 13 which shows that the “amazed” and “ashamed” posts were correctly identified only 3 and 4 times, and mistaken for the other classes in 62 and 49 cases, respectively. However, the well-

represented emotions have not been confused with the less frequent "amazed", "ashamed", "disgusted", and "pensive" classes in any instances. Thus, the imbalance in the corpus biases the model towards frequent emotions, and we attempt to address this issue by exploring the use of a weighted loss function and over-sampling methods in the next subsections.

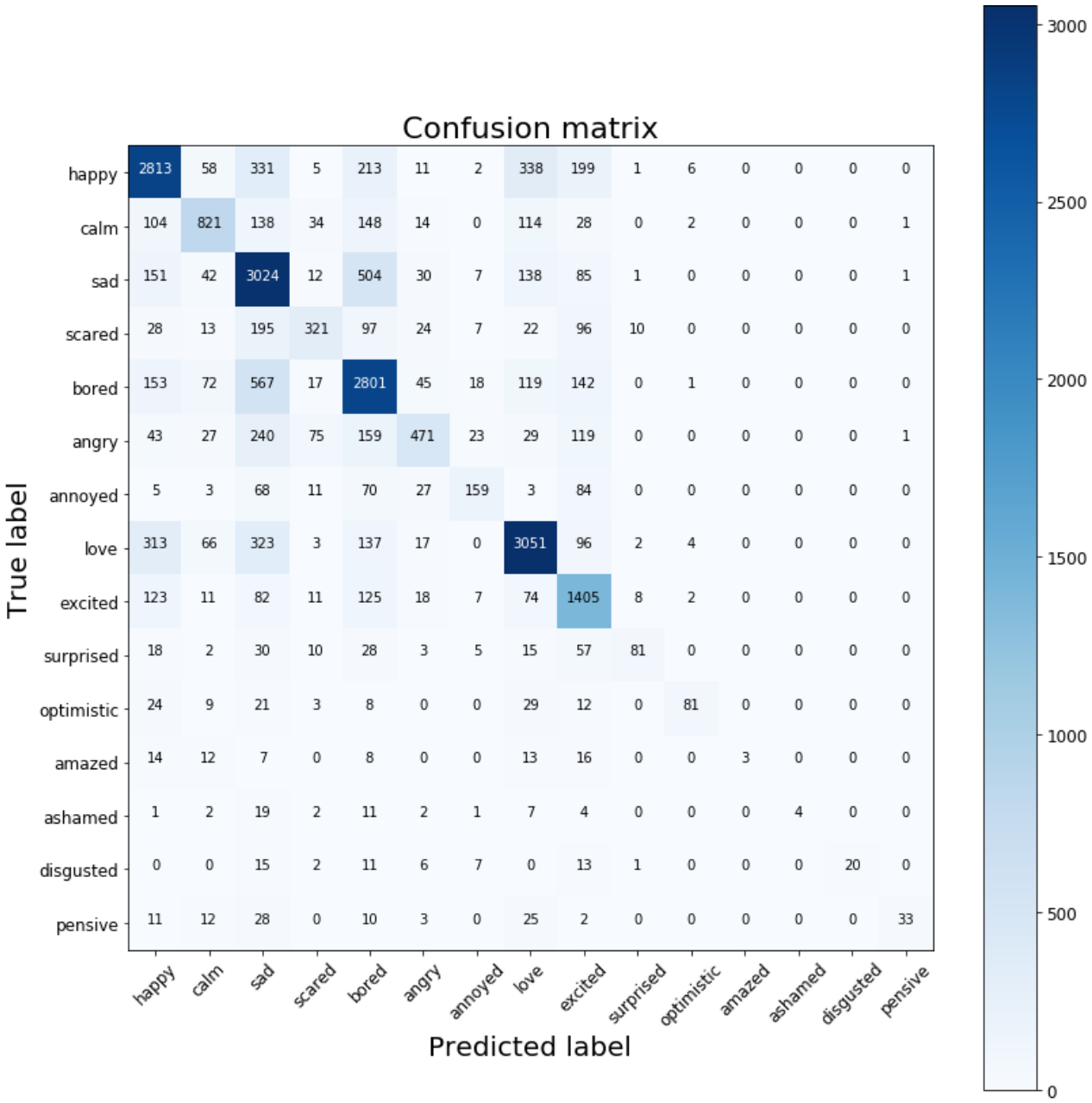

Figure 13: Confusion matrix for textual emotion classification **without class balancing**. The underlying model comprises word embeddings followed by a single LSTM layer of 1024 units.

6.1.3.2 Weighted loss

The categorical cross-entropy objective function used in the previous experiments backpropagates the error gradients for all classes with equal weight. In the case of our unbalanced dataset, this tunes the network towards the frequent classes as their corresponding errors will dominate the loss of the under-represented categories. Hence, the model fails to learn the characteristics of the infrequent emotions and mistakes them for the popular affects as discussed in the previous experiment.

Cost-sensitive classifiers address this shortcoming of standard cost functions by applying class-dependent weights to the error gradients. Thus, we can improve the attention of the model towards the infrequent emotions by applying a larger penalty to misclassifications of the minority classes than to errors corresponding to the frequent emotions [43]. The inverse of class support represents such a weighting scheme that applies tiny updates to the losses of well-represented classes and scales up the errors on infrequent categories. The weights are computed using the formula $w_c = \frac{1}{f_c}$ i.e. the weight for the gradient of a class is inversely proportional to the frequency of that class. This approach renders **75.3% training and 64.53% validation accuracy**, with table 5 showcasing further per-class metrics. While overfitting is evident from the accuracy scores, the prominent decrease of the recall, precision and F-1 scores in frequent emotions also suggests that the model is biased towards the features of the under-represented classes. Hence, the generalisation ability of the network on previously unseen instances of the popular affects is under-performant.

Emotion class	**Samples**	**Recall**	**Precision**	**F1-score**
Happy	40,003	0.67	0.75	0.71
Calm	14,325	0.68	0.57	0.62
Sad	40,003	0.60	0.68	0.64
Scared	8,156	0.54	0.42	0.47
Bored	39,692	0.63	0.68	0.66
Angry	11,844	0.51	0.45	0.48
Annoyed	3,965	0.62	0.29	0.39
Love	40,005	0.69	0.80	0.74
Excited	19,163	0.62	0.73	0.67
Surprised	2,579	0.48	0.26	0.34
Optimistic	1,868	0.64	0.33	0.43
Amazed	749	0.33	0.11	0.17
Ashamed	477	0.34	0.10	0.15
Disgusted	670	0.53	0.17	0.25
Pensive	1,106	0.52	0.34	0.41

Table 5: Precision, recall and F1-score metrics achieved by a single LSTM layer of 1024 units using a **weighted loss function** on the Tumblr dataset.

The trend outlined by the precision and F1 curves in graph 14 - a proportional increase with the number of samples per class - resembles the pattern obtained by training without a class balancing method (depicted in figure 12). However, while the trends of the two graphs are similar, the values of the precision and F1 scores decreased as a result of weighting the errors. The confusion matrix in figure 15 supports this observation as it shows that while slightly more instances of under-represented classes are now correctly classified, a massive amount of misclassifications occurs on the frequent emotions as they are mistaken for the rare ones.

This situation is the reverse of training without class-balancing from the previous experiment, where under-represented emotions were mistaken for popular ones. The shift in the ability of the model to predict only the relevant posts for the frequent emotions causes the recall on the infrequent classes to significantly drop too. This explains the inverted trend followed by the recall, which now increases with the number of samples.

The conclusion that can be drawn from these results is that rather than improving the performance of the network on the infrequent emotions, the inverse class-support scheme confuses the model as it performs poorer on the frequent classes, while no significant improvement can be observed for the rare affects. The line of weighted loss schemes was not further pursued as it does not represent a promising avenue.

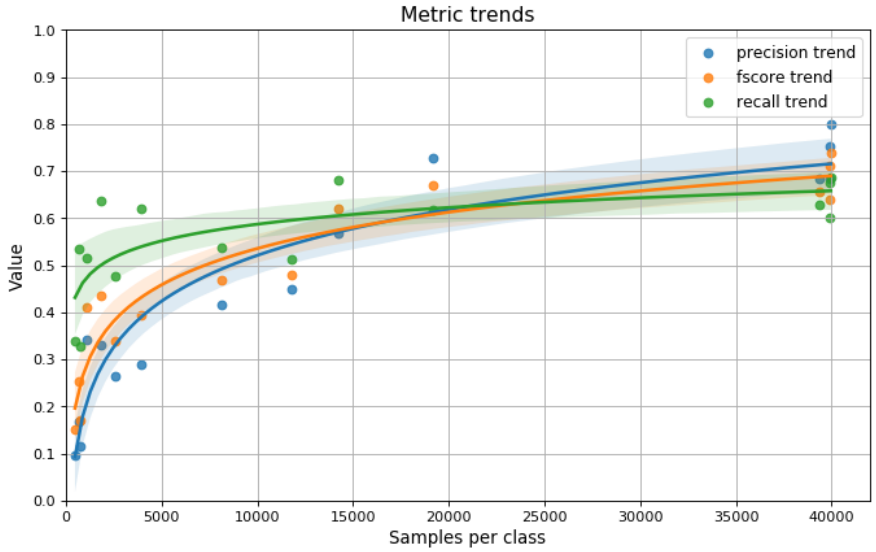

Figure 14: Precision, recall and F1 trends over the number of samples per emotion class for **inverse class-support weighted loss function**.

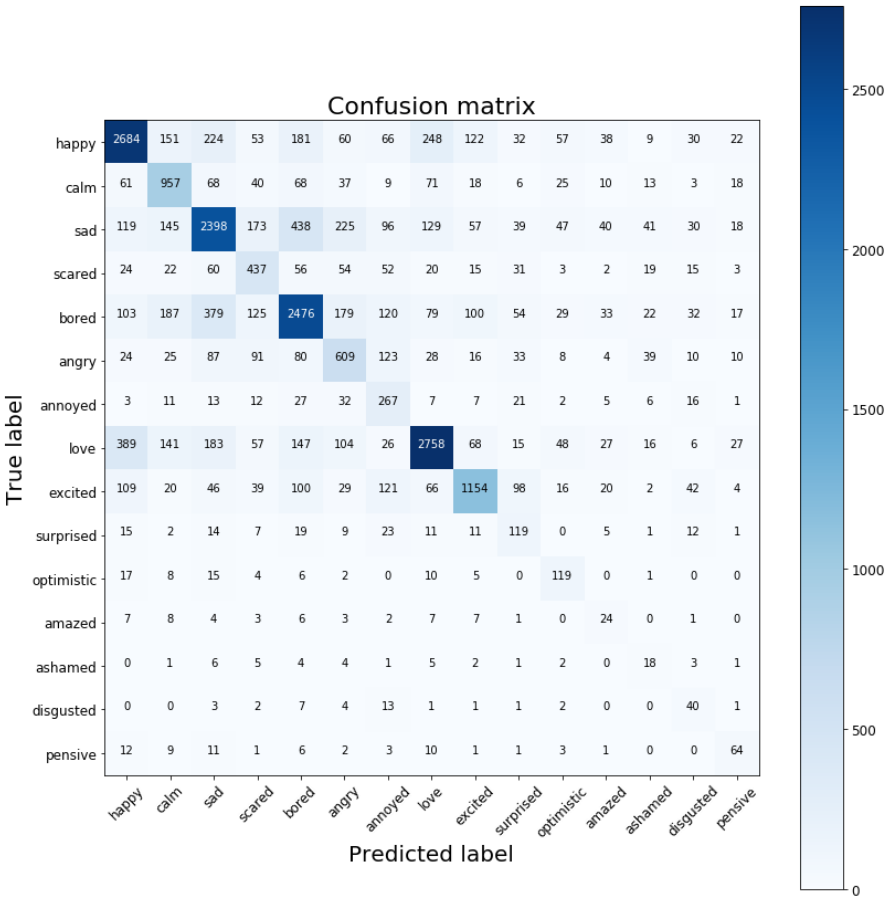

Figure 15: Confusion matrix for **inverse class-support weighted loss function**.

6.1.3.3 Random over-sampling

Random over-sampling duplicates samples of the minority classes in a stochastic manner in order to balance the distribution of records across our emotion categories. This ensures that the same number of textual descriptions will be presented to the model for each class during a training epoch. However, this naive sampling with replacement approach raises the chances of overfitting since the network is fed with the same examples of the under-represented classes repeatedly and can end up memorising instead of learning their characteristics.

We report **88.03% training and 66.01% validation accuracy** as the result of this experiment. The difference between the two scores clearly indicates overfitting, significantly exceeding the margin obtained without class balancing. Although the validation accuracy does not represent a new best, we observe improvements in the metrics depicted in table 6 below. While the precision and F1 scores slightly decreased in the case of frequent emotions compared to training without class balancing methods, they have increased for under-represented emotions such as "amazed", "ashamed", and "pensive". However, this still comes at the cost of frequent emotions being mistaken for rare ones (confusion matrix in figure 17), even though these errors are not as prominent as in the case of weighted loss, and causes the recall of infrequent classes to significantly drop. The precision and F1 trends are not different from the previous experiments and depict a continuous increase with the number of samples per class (figure 16).

Emotion class	**Samples**	**Recall**	**Precision**	**F1-score**
Happy	40,003	0.69	0.77	0.73
Calm	14,325	0.64	0.61	0.63
Sad	40,003	0.66	0.64	0.65
Scared	8,156	0.54	0.42	0.47
Bored	39,692	0.64	0.70	0.67
Angry	11,844	0.51	0.51	0.51
Annoyed	3,965	0.54	0.43	0.48
Love	40,005	0.76	0.75	0.75
Excited	19,163	0.64	0.71	0.67
Surprised	2,579	0.44	0.46	0.45
Optimistic	1,868	0.55	0.60	0.57
Amazed	749	0.29	0.43	0.34
Ashamed	477	0.25	0.16	0.20
Disgusted	670	0.64	0.09	0.16
Pensive	1,106	0.49	0.62	0.55

Table 6: Precision, recall and F1-score metrics achieved by training with **random over-sampling** on the Tumblr dataset.

Although applying random over-sampling resulted in a minor decrease in the overall accuracy, this line of methods provides a promising avenue to handling the imbalance issue of our corpora. Hence, in the next section, we apply a more complex over-sampling technique, namely SMOTE, to synthesise new text sequences from the available samples rather than simply duplicating them.

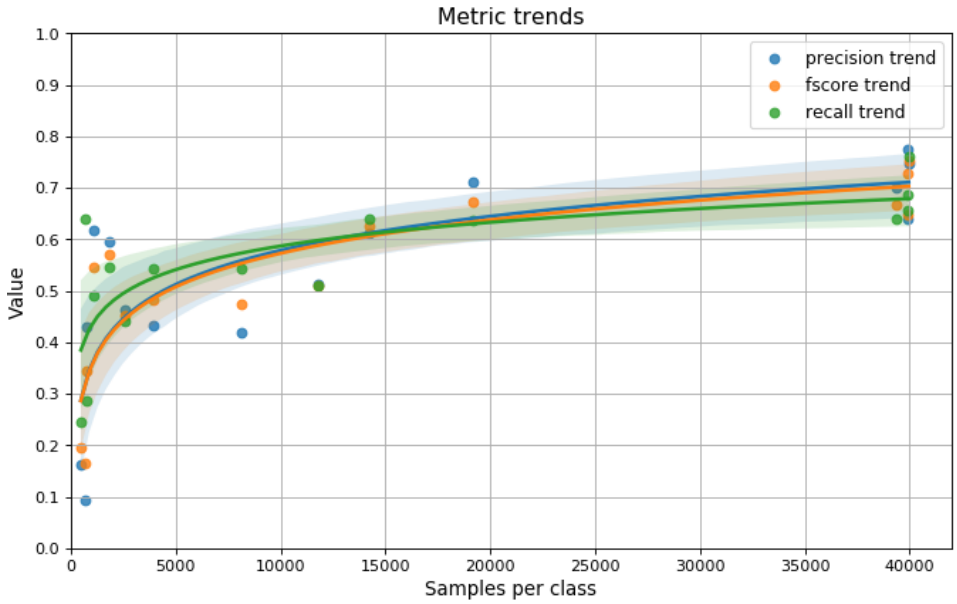

Figure 16: Precision, recall and F1 trends over the number of samples per emotion class for **random over-sampling**.

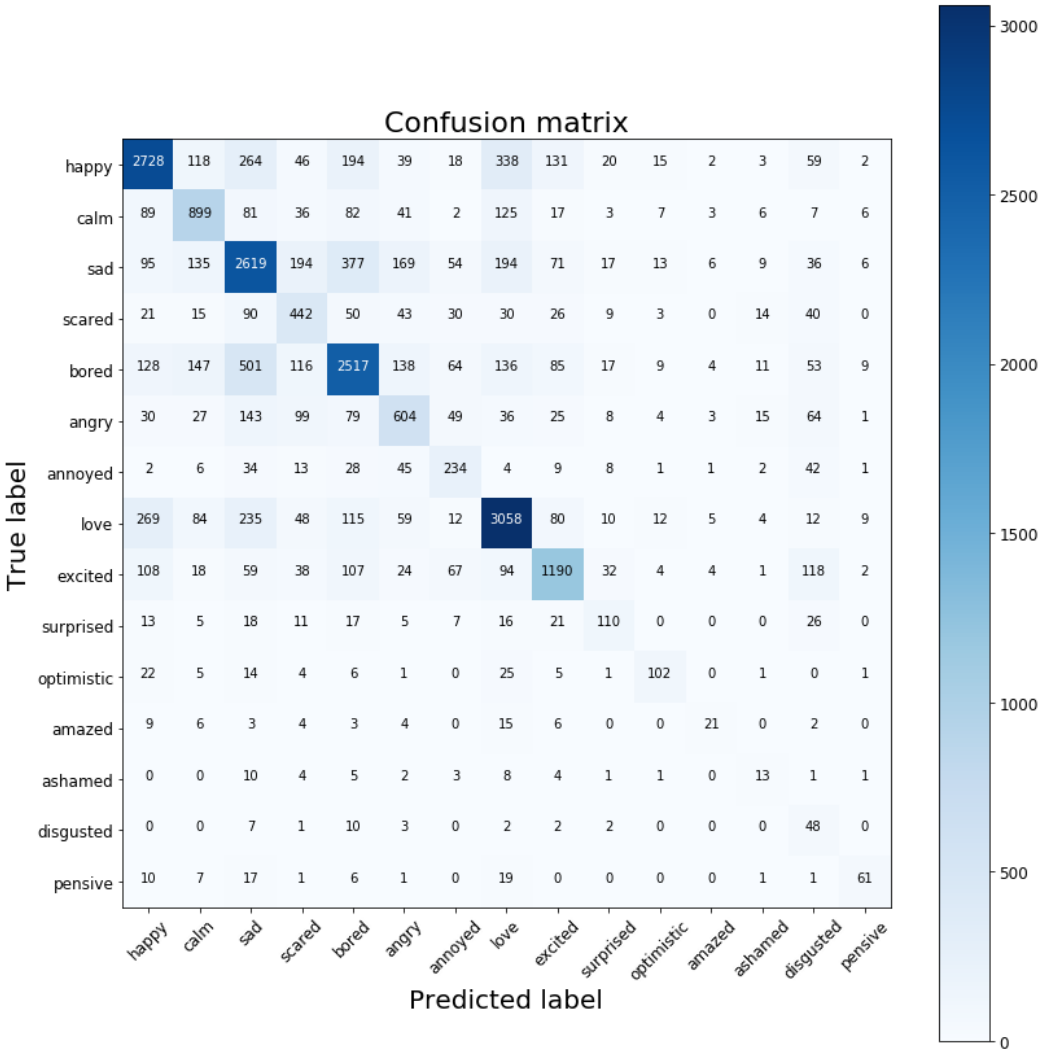

Figure 17: Confusion matrix for training with **random over-sampling**.

6.1.3.4 SMOTE over-sampling

SMOTE (Synthetic Minority Oversampling Technique) synthesises additional samples for the minority classes. Data points belonging to these under-represented classes are randomly selected and a k-nearest neighbour algorithm applies on these points produces new training examples [44]. This achieves an increased variation among the samples of the infrequent classes and reduces the risk of overfitting as in the case of naive random over-sampling.

The augmentation is applied on the integer encoded words rather than on the word vectors as the word embeddings layer is fine-tuned during training and its weights change continuously. Even though this approach did not promise any results as the integer encodings of the words do not capture semantic relationships, the obtained performance exceeds that of random over-sampling. The achieved **71.62% training and 67.82% validation accuracies** come at the top of this series of class balancing experiments, although still below training without any such techniques. The amount of overfitting has been considerably reduced compared to the previous methods. Per-class metrics are showcased in table 7 and the confusion matrix is depicted in figure 19.

Emotion class	**Samples**	**Recall**	**Precision**	**F1-score**
Happy	40,003	0.70	0.76	0.73
Calm	14,325	0.63	0.69	0.66
Sad	40,003	0.72	0.61	0.66
Scared	8,156	0.51	0.55	0.53
Bored	39,692	0.70	0.67	0.68
Angry	11,844	0.48	0.68	0.56
Annoyed	3,965	0.45	0.77	0.57
Love	40,005	0.74	0.81	0.77
Excited	19,163	0.64	0.77	0.70
Surprised	2,579	0.67	0.22	0.33
Optimistic	1,868	0.68	0.77	0.72
Amazed	749	0.56	0.67	0.61
Ashamed	477	0.43	0.26	0.33
Disgusted	670	0.55	0.25	0.34
Pensive	1,106	0.70	0.32	0.44

Table 7: Precision, recall and F1-score metrics achieved by training with **SMOTE** on the Tumblr dataset.

SMOTE produces the highest F1 scores across all experiments, thus achieving a performance balance between precision and recall. This supports the data in the confusion matrix as fewer frequent emotions are mistaken for rare affects, which also explains the rise in accuracy. We can observe that this trade-off has been constantly improved throughout this series of experiments. The overall trends for the precision, recall and F1 metrics remains unchanged, however, the achieved values situate higher on the vertical axis. Hence, SMOTE helped the model learn the characteristics of the under-represented classes while not prominently biasing its ability in predicting the frequent ones.

We conclude this section with the finding that SMOTE must be applied for balancing the distribution of classes used by DeepSentiment's emotion model in order to improve the performance of the network in predicting the entire spectrum of affects. Otherwise, the model will focus only on the frequent categories and fail to learn the characteristics of the rare ones.

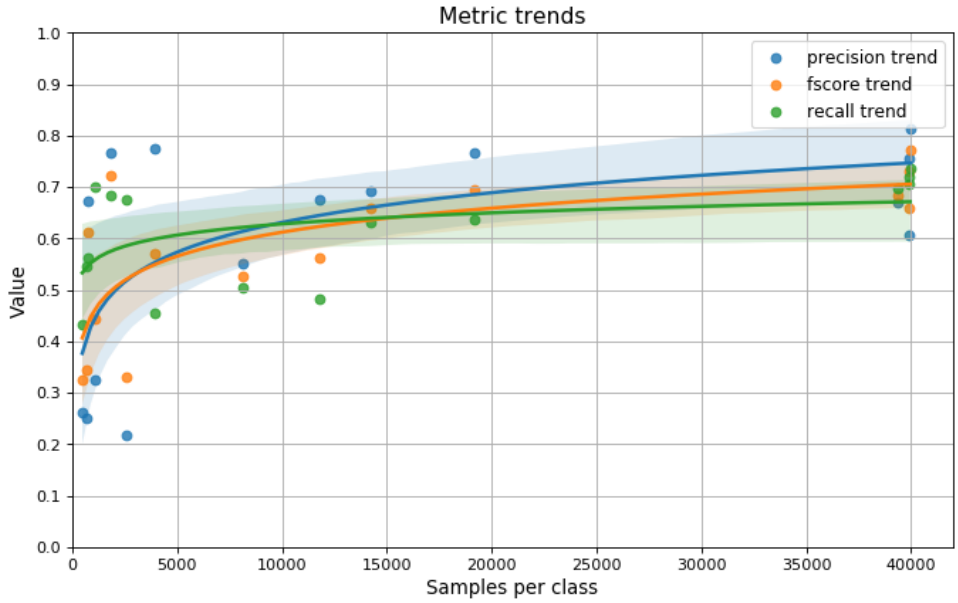

Figure 18: Precision, recall and F1 trends over the number of samples per emotion class for training with **SMOTE**.

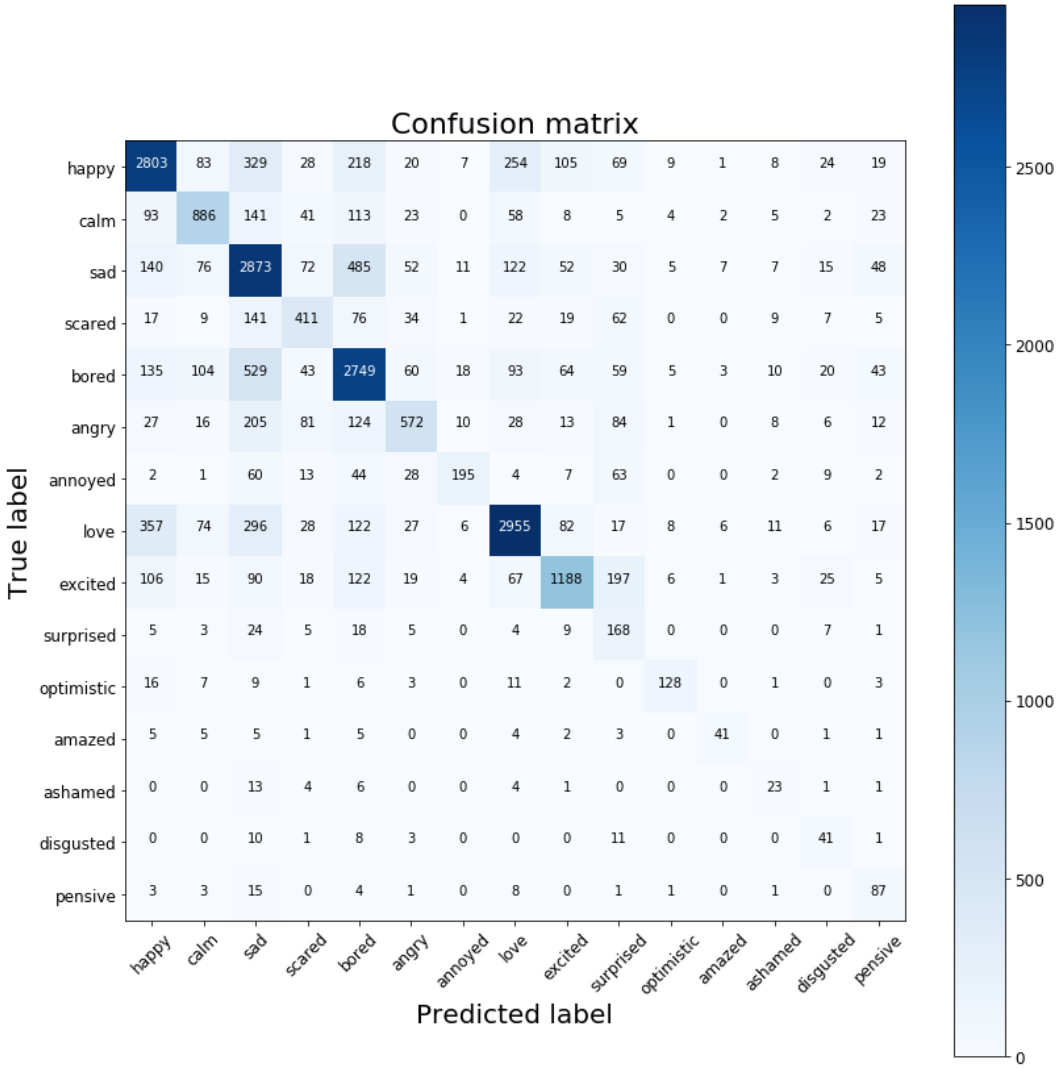

Figure 19: Confusion matrix for training with **SMOTE**.

6.1.4 Performance comparison on the Twitter and Tumblr datasets

The appositeness of Tumblr and Twitter to the task of textual emotion recognition is evaluated by training our model on both corpora. We use the basic emotion model as the frame of reference since the Twitter dataset lacks almost entirely the additional affects encompassed by DeepSentiment's emotion spectrum, such as "ashamed" or "pensive". However, the large gap in the number of samples per class among emotions from the basic model in the corpus of tweets further questions the ability of the network to grasp the under-represented affects from this dataset; the discrepancy between the 15,863 "happy" and "240" surprised posts, respectively, is massive. Hence, applying class balancing techniques would only cause overfitting on these few samples, so training is carried out without handling this imbalance. On the other hand, the Tumblr corpus has a dense distribution across the emotions from the basic model, with "disgusted" being the only outlier, standing at 670 samples. The accuracy results are showcased in table 8 below.

Dataset	**Total size**		**Accuracy**	
	Training	Validation	Training	Validation
Twitter	16,010	1,779	98.59%	**97.63**%
Tumblr	92,659	10,296	89.35%	**79.22**%

Table 8: Training and validation accuracies achieved by the textual emotion recognition model on the Tumblr and Twitter datasets using the **basic emotion model.**

The impressive validation accuracy of 97.63% obtained on the Twitter corpus is supported by the per-class metrics depicted in table 14. We can observe that the classifier is not biased towards the predominant "happy" emotion and it successfully learned the characteristics of the other affects too. The confusion matrix in figure 20 shows that the rare classes are correctly classified in the majority of cases, with only "happy" and "sad" being mistaken for each other on a few occasions. **We infer from these results that the conciseness of the Twitter posts helps in communicating the emotions more clearly and directly as we achieve state-of-the-art performance on a relatively small dataset of tweets.**

Emotion class	**Samples**	**Recall**	**Precision**	**F1-score**
Happy	40,003	0.99	0.98	0.99
Sad	40,003	0.72	0.73	0.72
Angry	11,844	0.71	1.00	0.83
Scared	8,156	0.94	1.00	0.97
Surprised	2,579	0.89	0.96	0.92
Disgusted	670	1.00	1.00	1.00

Table 9: Precision, recall and F1-score metrics achieved by the textual emotion recognition model on the **Twitter dataset** using the **basic emotion model**.

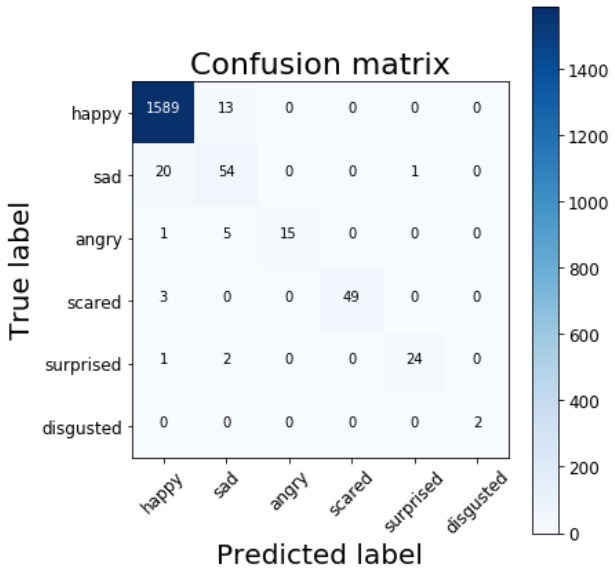

Figure 20: Confusion matrix on the **Twitter dataset** using the **basic emotion model**.

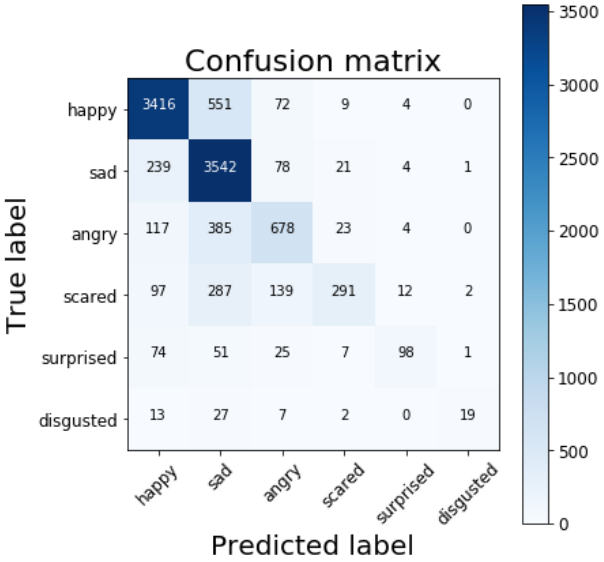

Figure 21: Confusion matrix on the **Tumblr dataset** using the **basic emotion model**.

The validation accuracy of 79.22% achieved by the textual emotion recognition network on the Tumblr dataset using the basic spectrum of six emotions exceeds by over 10% the accuracy obtained on DeepSentiment's emotion model. While the absence of extremely rare affects such as "ashamed" and "amazed" is an important factor in this improvement, the model is biased towards the majority classes and still mistakes the infrequent emotions for the popular ones (confusion matrix in figure 21). This is supported by the low recall obtained on the under-represented classes - 0.35 for "scared", 0.38 for "surprised", and 0.28 for "disgusted" (table 14). **The key takeaway from these results is that the expressiveness of the Tumblr posts and the subtle manner in which affects are communicated makes the emotion recognition task more difficult to the classifier. Furthermore, there must exist a partial overlap across the expression forms of the affects in the studied emotion spectra, which complemented by the imbalance in our dataset, encourages the model to always pick the common emotions.**

Emotion class	Samples	Recall	Precision	F1-score
Happy	40,003	0.84	0.86	0.85
Sad	40,003	0.91	0.73	0.81
Angry	11,844	0.56	0.68	0.61
Scared	8,156	0.35	0.82	0.49
Surprised	2,579	0.38	0.80	0.52
Disgusted	670	0.28	0.83	0.42

Table 10: Precision, recall and F1-score metrics achieved by training the textual emotion recognition model on the **Tumblr dataset** using the **basic emotion model**.

6.1.5 Knowledge transferability across Twitter and Tumblr

The transferability of the learned knowledge across the two social platforms is assessed by evaluating the model trained on Tumblr dataset using its corpus of tweets counterpart and vice-versa. This approach should also indicate the degree of similarity of the textual expression of affects between the two networks. The accuracy results obtained are presented in table 11 below.

Dataset		**Accuracy**
Training	Testing	
Twitter	Tumblr	**38.59**%
Tumblr	Twitter	**88.13**%

Table 11: Cross-testing accuracies on the Tumblr and Twitter datasets using the **basic emotion model.**

The 38.59% accuracy achieved by the Twitter trained model on the Tumblr validation set indicates that the affective features learned on the corpus of tweets do not generalise to the more expressive Tumblr posts. This may be due to the discrepancy in richness between the vocabularies of the two social network platforms. While the Twitter corpus contains only 27,584 unique words, its Tumblr counterpart stands at 225,542. Hence, a large number of lexemes present in the Tumblr posts do not have corresponding word vectors learned by the model trained on Twitter data and are mapped to Out-of-Vocabulary (OOV) tokens. The confusion matrix in figure 26 support these claims, with all samples of the Tumblr dataset being predicted as "sad". On the other hand, the model trained on the Tumblr corpus which is biased towards the frequent emotions classifies every tweet as "happy"; thus, the high accuracy score of 88.13% is due to the prominence of the "happy" emotions in the Twitter dataset.

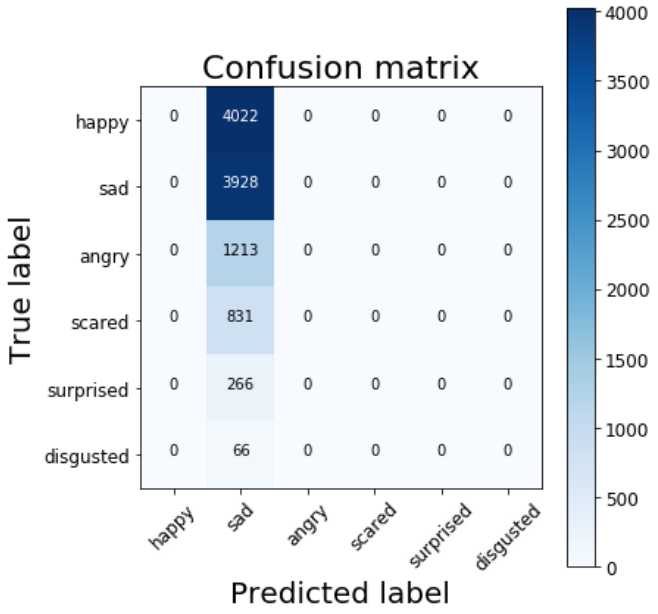

Figure 22: Confusion matrix obtained by evaluating the **Twitter trained model** on the **Tumblr dataset** using the **basic emotion model**.

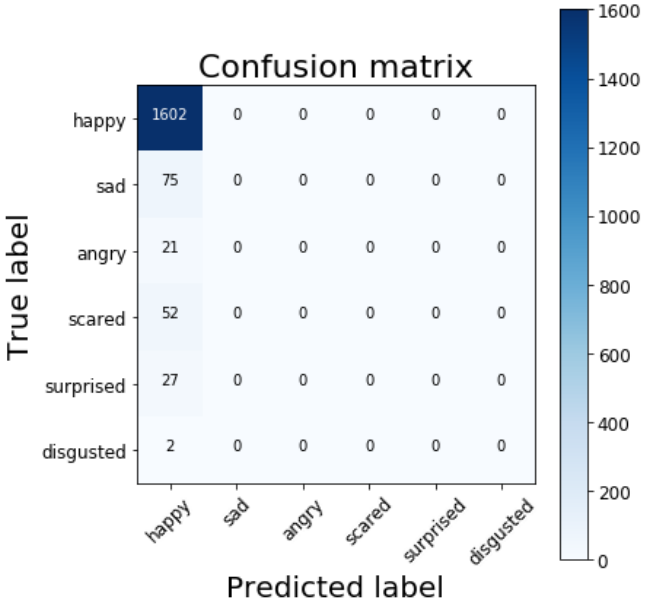

Figure 23: Confusion matrix obtained by evaluating the **Tumblr trained model** on the **Twitter dataset** using the **basic emotion model**.

We conclude with the observation that while a model jointly trained on data collected from multiple social networks may capture general characteristics of affects shared across several platforms, the high variation in forms of expression across these media makes it difficult for a single model to capture all peculiar features specific to individual platforms. Hence, the optimal approach to solving the textual emotion recognition for all social networks is to build a framework of models each of which has to be trained separately on data belonging to a single platform.

6.2 Visual Emotion Analysis

The automatic detection of emotions from images alone requires the ability to capture complex representations at a higher level of abstraction than other known problems such as image segmentation or classification [8]. We explore the inherent difficulty of the problem by performing transfer learning on the InceptionV3, a state-of-the-art CNN for image recognition, in order to assess the expressiveness of the visual component in conveying emotions. The scarce number of samples across most classes in the Twitter dataset makes it unsuitable for a Deep Learning approach as large amounts of data are needed to optimise the top 4 million parameters of Inception. Hence, we fine-tune InceptionV3 on the Tumblr dataset using both the basic and DeepSentiment's emotion models and compare the results. The accuracy scores achieved on these spectra are presented in table 12 below. Furthermore, we explore data augmentation techniques such as rotations, width and height shifts, zooming, and horizontal flips which are applied to the pictures and report on the improvements.

Emotion model	**No augmentation**			**Augmentation**		
	Training	**Validation**	Time	Training	**Validation**	Time
Basic	81.44%	**62.03%**	2h 15m	64.73%	**62.61%**	5h 9 m
DeepSentiment	50.46%	**41.35%**	5h 37m	43.41%	**40.97%**	8h 17m

Table 12: Training and validation accuracies obtained by fine-tuning InceptionV3 on the basic and DeepSentiment's emotion models with and without data augmentation.

6.2.1 Performance on DeepSentiment's emotion model

The validation accuracy of 41.35% obtained on DeepSentiment's emotion spectrum exceeds by over 5% the accuracy of DeepSentiment on the visual emotion recognition task. However, this statistic is still not brilliant and proves that even state-of-the-art CNNs have difficulties in grasping the underlying structure of complex emotions in images. For example, underrepresented classes such as "amazed' and "ashamed" have corresponding precision and recall values of 0 (metrics table 13) as they are incorrectly classified in all instances (confusion matrix in figure 24). Although frequent classes such as "happy", ""sad", "bored" and "love" achieve higher per-class metric scores, the model confuses even these affects with one another.

Data augmentation methods doubled the training time but did not bring any benefits since the accuracy slightly dropped and the degree of confusion across emotions has increased. The number of true positives for rare affects has decreased even more along with the F1 scores (confusion matrix for data augmentation in figure 25), with the model focusing on a small group of classes comprising "love", "bored", and "sad" as the rest of the emotions are commonly mistaken for these.

		No augmentation			Augmentation		
Emotion class	**Samples**	**Recall**	**Precision**	**F1-score**	**Recall**	**Precision**	**F1-score**
Happy	40,003	0.43	0.35	0.38	0.25	0.38	0.30
Calm	14,325	0.24	0.59	0.34	0.26	0.57	0.36
Sad	40,003	0.69	0.45	0.55	0.64	0.49	0.55
Scared	8,156	0.11	0.61	0.18	0.13	0.43	0.20
Bored	39,692	0.45	0.46	0.46	0.42	0.49	0.46
Angry	11,844	0.14	0.46	0.21	0.19	0.29	0.23
Annoyed	3,965	0.21	0.79	0.33	0.15	0.52	0.23
Love	40,005	0.38	0.36	0.37	0.61	0.32	0.42
Excited	19,163	0.36	0.36	0.36	0.31	0.36	0.34
Surprised	2,579	0.16	0.51	0.24	0.04	0.32	0.07
Optimistic	1,868	0.12	0.81	0.21	0.12	0.80	0.20
Amazed	749	0.00	0.00	0.00	0.00	0.00	0.00
Ashamed	477	0.00	0.00	0.00	0.00	0.00	0.00
Disgusted	670	0.53	0.36	0.43	0.30	0.65	0.41
Pensive	1,106	0.20	0.88	0.32	0.22	0.62	0.32

Table 13: Precision, recall and F1-score metrics for **DeepSentiment's emotion model with and without data augmentation.**

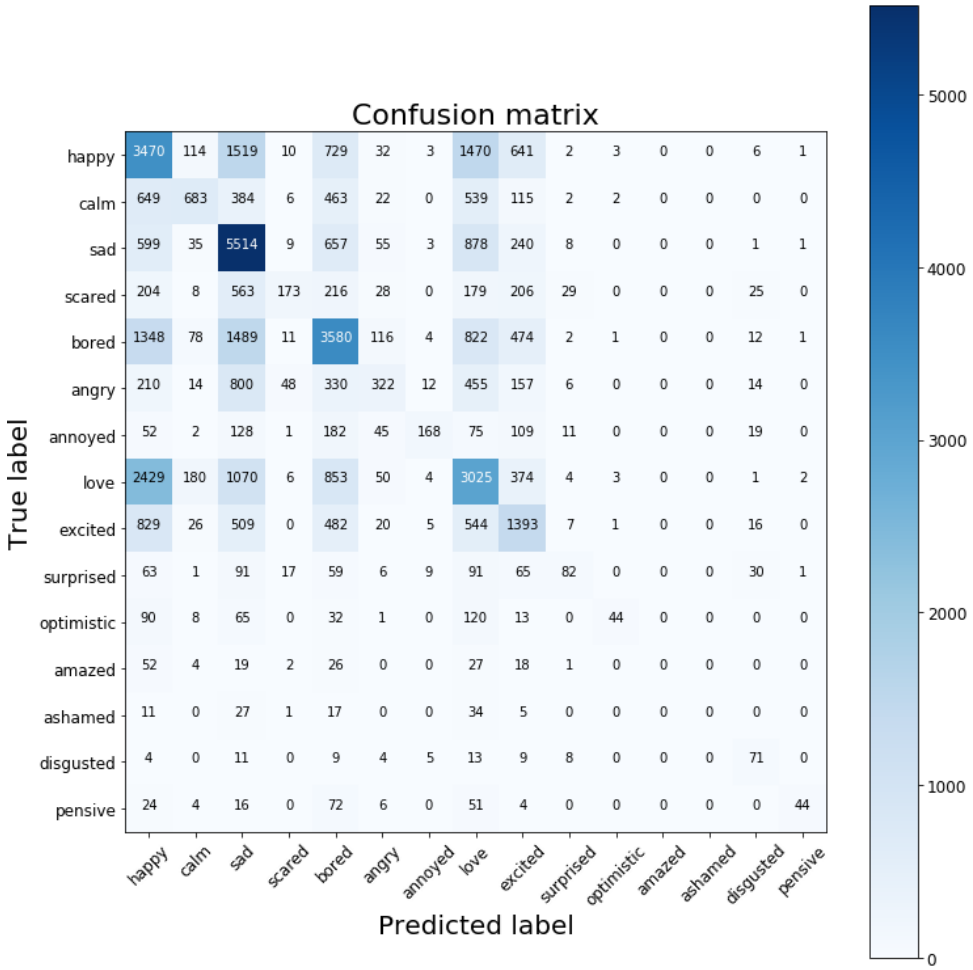

Figure 24: Confusion matrix for DeepSentiment's emotion model without data augmentation.

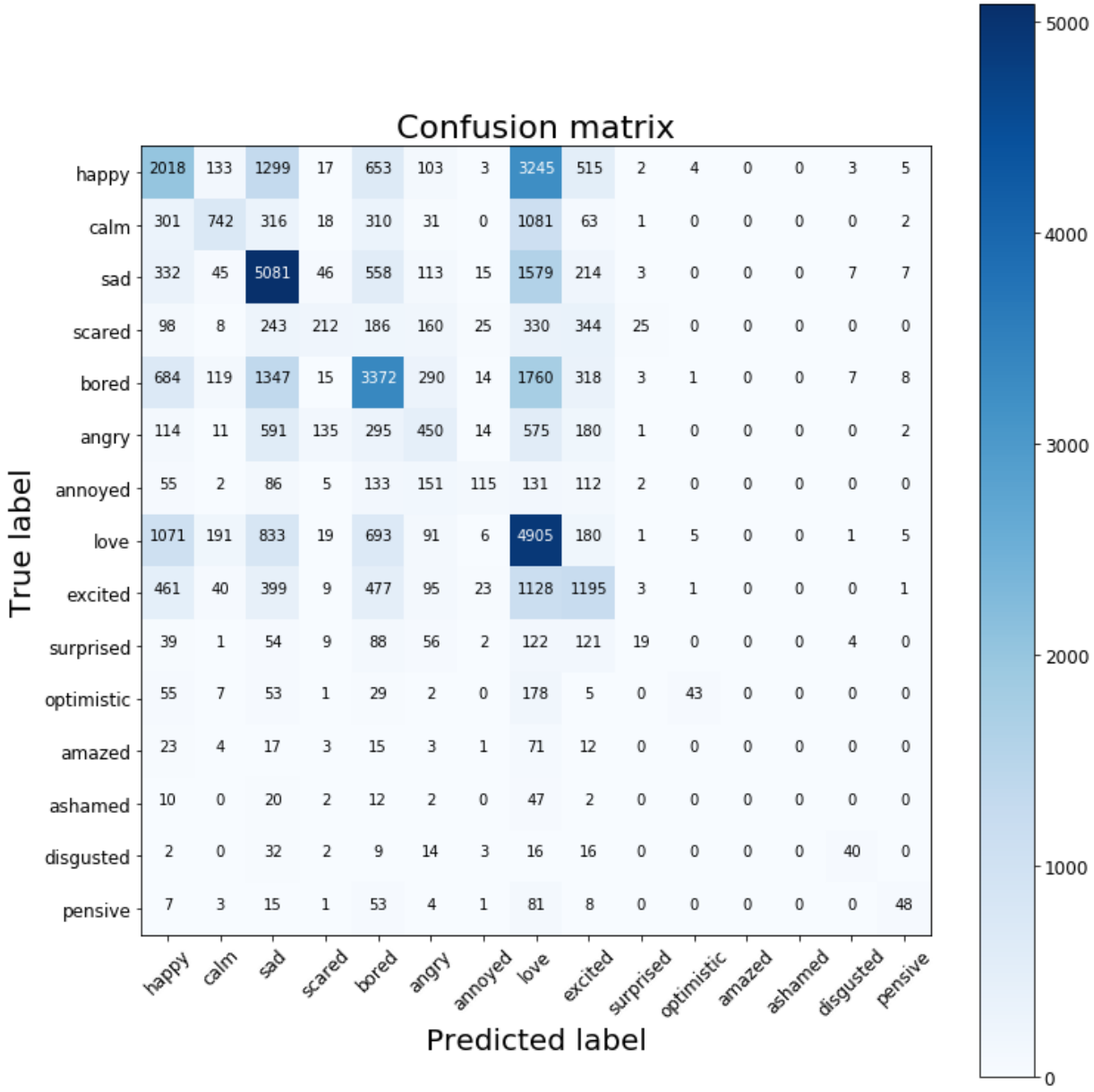

Figure 25: Confusion matrix for DeepSentiment's emotion model with data augmentation.

6.2.2 Performance on the basic emotion model

The 62.03% validation accuracy achieved on the basic emotion model without data augmentation represents a promising result which suggests that some underlying affective structure is present in the Tumblr images, and has been captured by the fine-tuned InceptionV3. However, looking further at the per-class metrics in table 13 we observe that all emotions except "happy" and "sad", which have been partially learned by the model, have low F1 scores. However, these results have been slightly improved by training with the data augmentation methods that apply geometrical transformations to the images. This is contrary to what has been obtained on DeepSentiment's emotion model and is due to the dense distribution of samples across the basic emotions in the Tumblr dataset. The relatively high F1 scores obtained for the "happy", "sad", and "disgusted" emotions of over 0.6 indicate that the network has learned to predict these classes reasonably well, while the attention towards "scared" and "surprised" has decreased by a minor margin. Hence, data augmentation proves beneficial when the corpus is sufficiently balanced across the encompassed classes. Finally, collecting more posts to perfectly balance the distribution of samples across all classes should render this approach a complete success.

Unfortunately, the Tumblr API has been exhausted as all existing posts for these emotions have been retrieved.

		No augmentation			Augmentation		
Emotion class	**Samples**	**Recall**	**Precision**	**F1-score**	**Recall**	**Precision**	**F1-score**
Happy	40,003	0.72	0.63	0.67	0.66	0.68	0.67
Sad	40,003	0.76	0.61	0.68	0.81	0.57	0.67
Angry	11,844	0.13	0.55	0.20	0.16	0.41	0.23
Scared	8,156	0.13	0.45	0.21	0.07	0.42	0.13
Surprised	2,579	0.15	0.30	0.20	0.15	0.24	0.18
Disgusted	670	0.23	0.84	0.36	0.49	0.77	0.60

Table 14: Precision, recall and F1-score metrics for the **basic emotion model with and without data augmentation.**

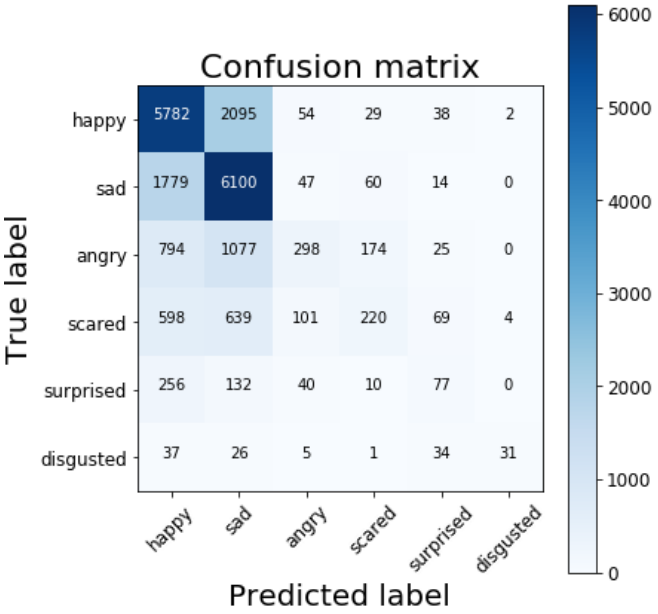

Figure 26: Confusion matrix for the **basic emotion model without data augmentation**.

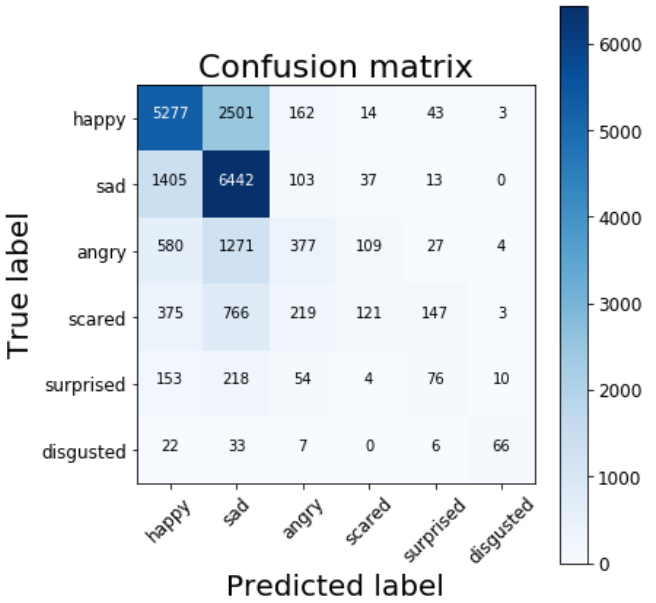

Figure 27: Confusion matrix for the **basic emotion model with data augmentation**.

The lower human performance on the task of emotion recognition from images compared to the solved object detection problem translated into the results obtained when transferring and optimising the knowledge of InceptionV3. This state-of-the-art CNN has achieved a 75.6% top-1 score on the 1000-class ILSVRC2015 image classification competition, however, after fine-tuning the network to predict an affect from a narrow spectrum of 6 emotions, this number has dropped to 62.61%. However, this result indicates that images contain a degree of structure that plays a major role in communicating the underlying emotion, with the optimised InceptionV3 capturing these affective patterns.

6.3 Multimodal Emotion Analysis

Social media text posts are often accompanied by images that complement the message and emphasize the communicated feeling. These mixed modalities can be accounted for by a multi-input network combining the textual and visual emotion recognition architectures evaluated in the previous sections. The outputs of the two modules are concatenated and a dense classifier produces a probability distribution over the selected emotion spectrum. We explore the performance of this model on both the basic and DeepSentiment's emotion models and report the accuracy results in table 15.

The Tumblr corpus is used only as the scarce number of samples across most classes in the Twitter dataset makes it unsuitable for a Deep Learning approach due to the x million parameters to be optimised in the multimodal network. Data augmentation and class balancing methods have been avoided in this experiment as mixing them on the two modalities could lead to confusing behaviours and a massive increase in training time.

Emotion model	**Training**	**Validation**	**Training Time**
Basic	87.74%	75.13%	2h 2m
DeepSentiment	74.70%	62.16%	8h 28m

Table 15: Training and validation accuracies obtained by the multimodal network on the basic and DeepSentiment's emotion models.

6.3.1 Performance on DeepSentiment's emotion model

The multimodal network of the OceanNet framework achieves a validation accuracy of 62.16% on the Tumblr dataset using DeepSentiment's emotion spectrum, which is 10% below the accuracy of DeepSentiment. The F1 scores in table 16 indicate that the model has learned to predict with good precision only a subset of emotions: "happy", "sad", "bored", "love", and "excited". The other affects are generally mistaken for the frequent ones as shown in the confusion matrix in figure 28, which resembles the behaviour encountered in the textual emotion analysis experiments. Furthermore, the values of the per-class metrics have decreased in comparison with the results of text-only emotion recognition. This leads us to the conclusion that the images merely confuse the network rather than raising the level of confidence in a prediction, and that text alone is sufficient in successfully classifying complex affects from social media.

Finally, the difference between OceanNet's and DeepSentiment's performance on the multimodal emotion recognition task is probably due to the nature of the gathered datasets. The amount of automatically generated blog posts has significantly increased in the recent Tumblr activity, and since we collected data that is up to 2 years more recent than those in DeepSentiment's dataset, we could notice that such artificially created posts have been retrieved as part of our corpus. A further area of research could focus on removing these from the dataset as they often contain mixed affective signals in the textual and visual modalities that confuse the network.

Emotion class	Samples	Recall	Precision	F1-score
Happy	40,003	0.64	0.64	0.64
Calm	14,325	0.58	0.47	0.52
Sad	40,003	0.77	0.59	0.66
Scared	8,156	0.35	0.54	0.43
Bored	39,692	0.52	0.74	0.61
Angry	11,844	0.41	0.35	0.38
Annoyed	3,965	0.45	0.37	0.40
Love	40,005	0.58	0.71	0.64
Excited	19,163	0.61	0.56	0.58
Surprised	2,579	0.35	0.23	0.28
Optimistic	1,868	0.30	0.21	0.25
Amazed	749	0.02	0.05	0.03
Ashamed	477	0.01	0.50	0.02
Disgusted	670	0.38	0.86	0.53
Pensive	1,106	0.34	0.17	0.22

Table 16: Precision, recall and F1-score metrics achieved by the multimodal model on the Tumblr dataset using **DeepSentiment's emotion model**.

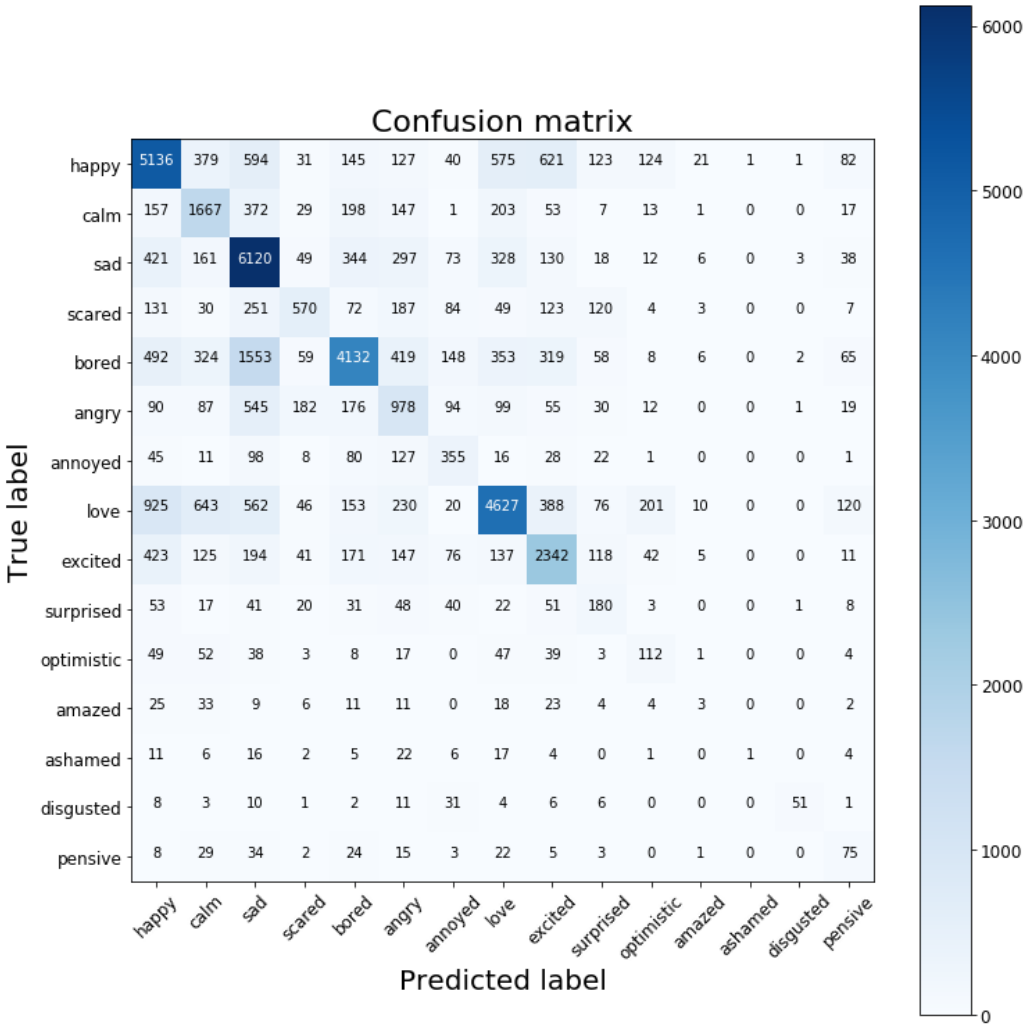

Figure 28: Confusion matrix for the multimodal network's results on **DeepSentiment's emotion model**.

6.3.2 Performance on the basic emotion model

The 75.13% validation accuracy of the multimodal network on the basic model of emotions represents a large improvement step from the results achieved on DeepSentiment's affective spectrum. The metrics in table 17 indicate that the model has learned to correctly predict the "happy" and "sad" posts as they have associated F1 scores of over 0.8, while performing generally well on the other classes too. The confusion matrix in figure 29 shows that rare affects are mistaken for the frequent emotions, however, the degree of confusion has decreased compared to the previous experiment. This is also supported by the increase in the values of the per-class metric scores. Nevertheless, the performance of the multimodal network on the basic emotions model is approximately 3% lower than for the text-only model, which reinforces the fact that images to not strengthen the confidence in a classification but rather confuse the network in the training process.

Emotion class	**Samples**	**Recall**	**Precision**	**F1-score**
Happy	40,003	0.84	0.86	0.85
Sad	40,003	0.91	0.73	0.81
Angry	11,844	0.56	0.68	0.61
Scared	8,156	0.35	0.82	0.49
Surprised	2,579	0.38	0.80	0.52
Disgusted	670	0.28	0.83	0.42

Table 17: Precision, recall and F1-score metrics achieved by the multimodal model on the Tumblr dataset using **the basic emotion model**.

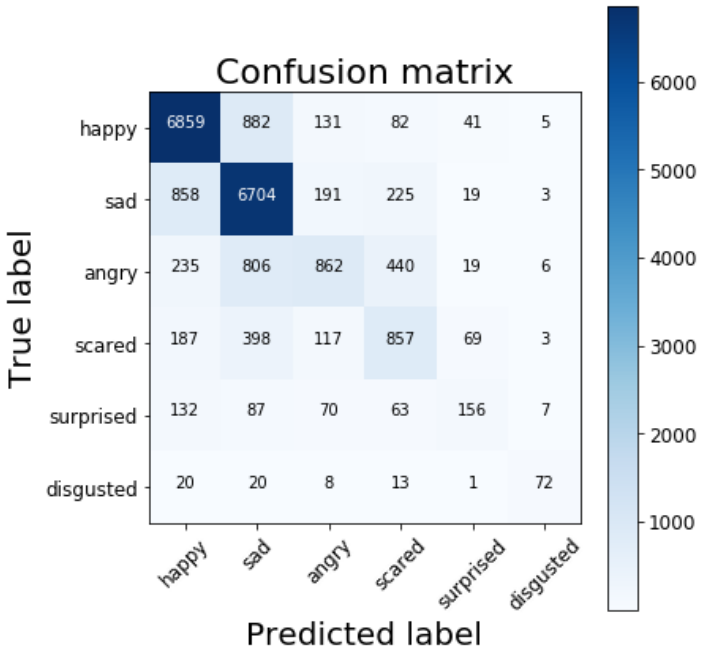

Figure 29: Confusion matrix for the multimodal network's results on the **basic emotion model**.

7 OceanNet - An Emotion Analysis Tool for Market Analysis Platforms

The models proposed for single- and multi-modal emotion recognition have been integrated into a novel framework, namely OceanNet, that provides a visualisation tool for market analytics platforms. Several dashboards have been developed as part of a full-stack web application that produces emotion statistics from image and text posts retrieved in real-time using a search functionality or uploaded by a sales person. The showcased data figures are computed from the results of feeding the neural networks with the Tumblr posts at hand. The predicted emotion labels can be visualised on the individual posts as well.

7.1 Presentation

The search functionality retrieves posts that contain the term of interest with the named-entity tag selected from the drop-down menu next to the search bar (figure 31). These posts are then fed through the single- and multi-modal networks of our framework and emotion distribution statistics are produced. Figure 30 showcases the results of a search for "Apple" over all named-entity tags.

The line graphs display the affective evolution trend of the search term on Tumblr over the last week by plotting and interpolating the number of posts classified for every emotion in each day. We can toggle between the statistics computed by the different neural networks and choose between textual, visual or multimodal analysis of the data using the top-right buttons. Figure 31 shows the multimodal statistics on the same "Apple" search term over all named-entity entity.

The bar charts that follow present the overall distribution of textual, visual and multimodal predictions over the emotion classes for all posts retrieved. We can observe from the "Apple" search that the three networks agree that most of the posts are either "happy" or "sad", with counts of over 50 each. This inter-model agreement strengthens the confidence in the visualised predictions. The colour scheme of the emotions has been selected from Plutchik's Wheel of Emotion [33].

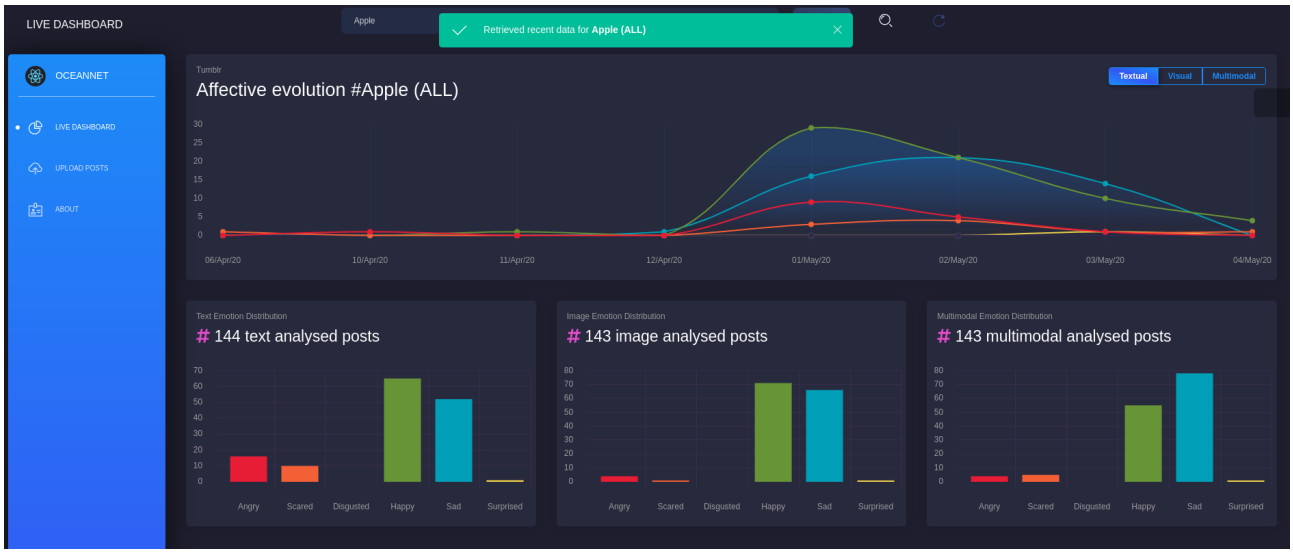

Figure 30: Real-time statistics for search term "Apple" over all named-entity tags. Affective evolution depicted for **textual classifications**.

Figure 31: Real-time statistics for search term "Apple" over all named-entity tags. Affective evolution depicted for **multimodal classifications**.

The retrieved posts with the highest confidence scores in the predicted emotion are showcased for all classes underneath the analytics frames. Figures 32 and 33 displays four top results for the "Happy" emotion fetched by an "Apple" search with the "Organisation" named-entity tag. The confidence scores for the most probable label are higher than 90%. However, we can observe that some non-English posts have been retrieved as they contain a large number of English hashtags. This suggests that a different filtering approach should be considered. Finally, this functionality allows market analysts to process posts individually and take action based on the opinions expressed by users.

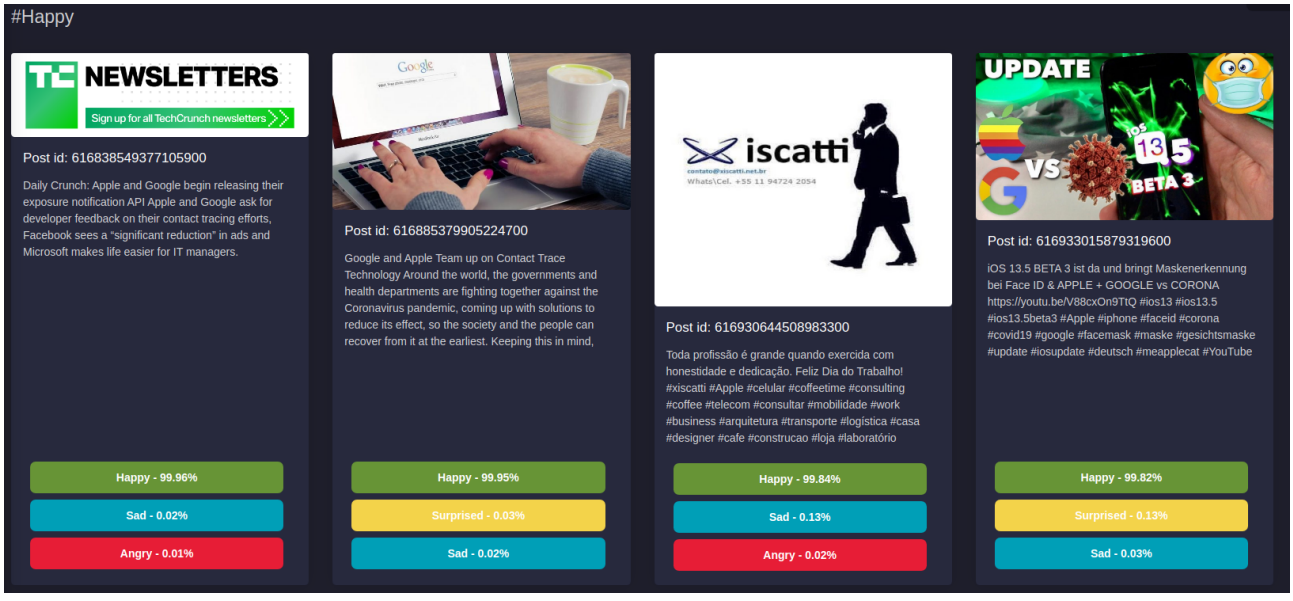

Figure 32: Posts classifications for search term "Apple" with "Organisation" named-entity tag.

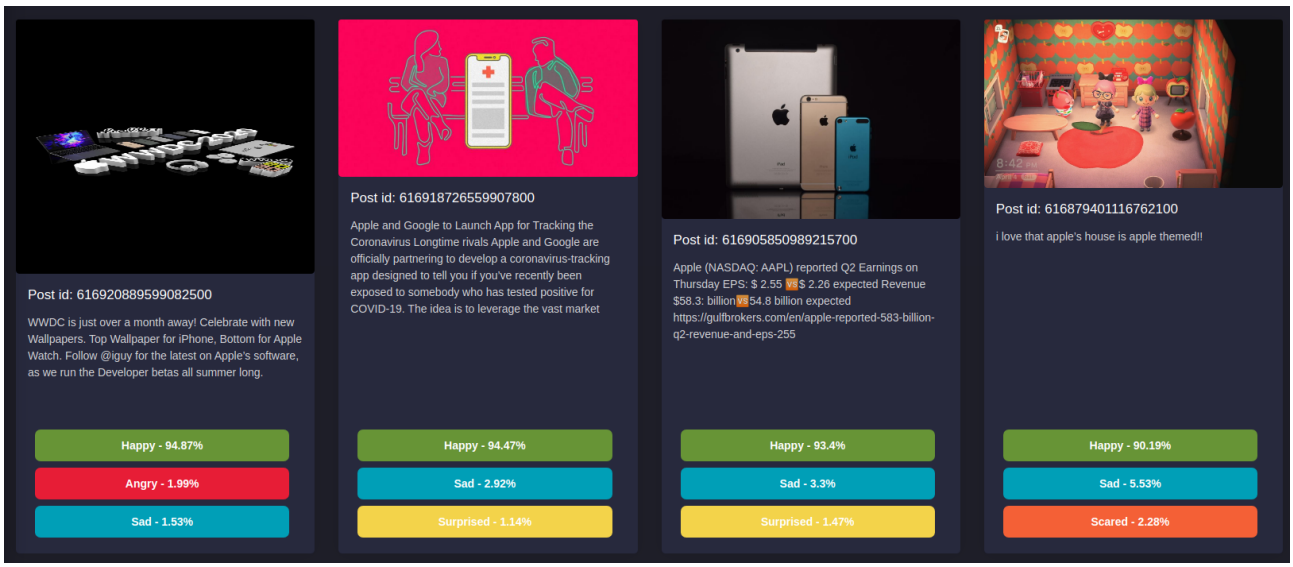

Figure 33: Posts classifications for search term "Apple" with "Organisation" named-entity tag.

Collections of Tumblr data already gathered by sales analysts can also be uploaded for classification as CSV files that contain the textual description and image URL of each post. Furthermore, the user can select the modalities to be accounted for in the prediction. The interface for uploading and selecting the network is depicted in figures 34 and 35. Collections of randomly selected Tumblr posts have been uploaded for analysis and the statistical results are shown in figure 36, while concrete predictions are depicted in figure 37.

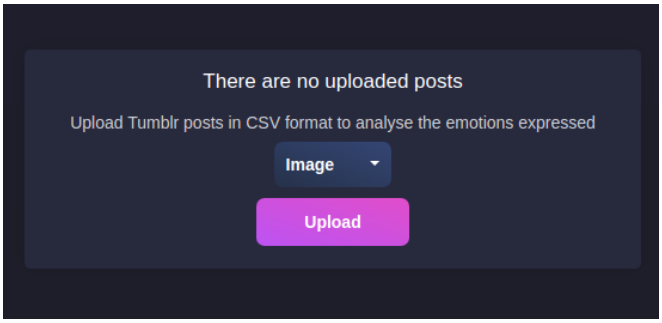

Figure 34: Interface for uploading posts.

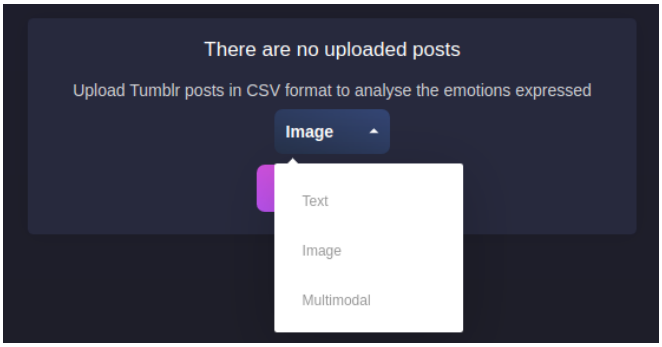

Figure 35: Model selection for uploading posts.

Figure 36: Distribution statics for uploaded posts.

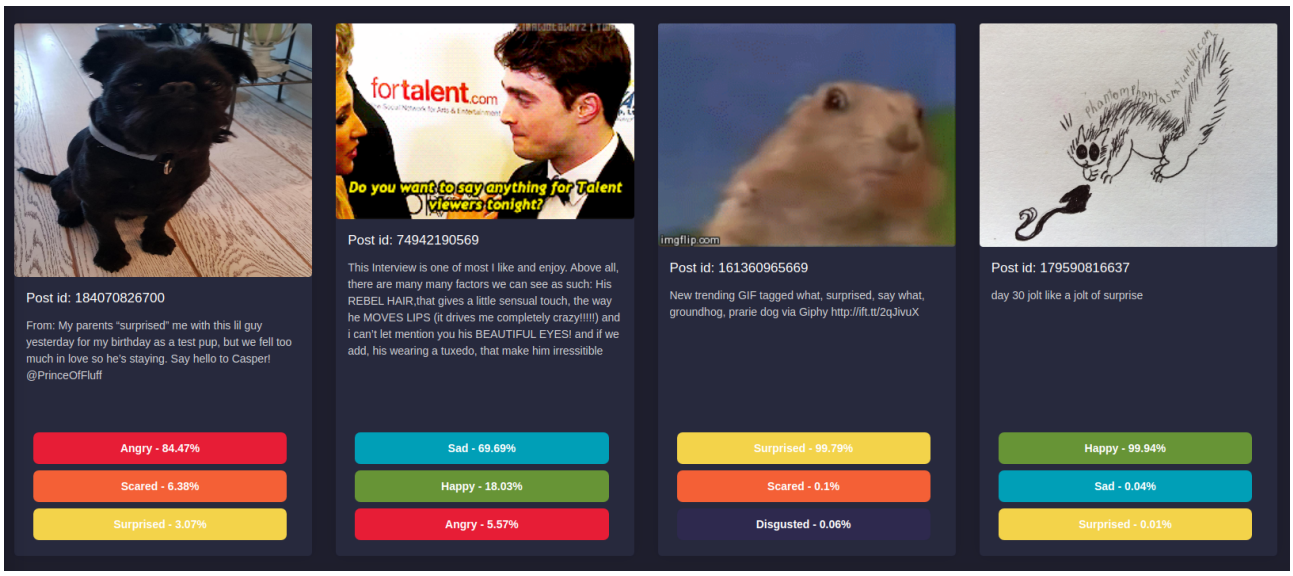

Figure 37: Predictions for uploaded posts.

7.2 Implementation

The visualisation system of OceanNet is a full-stack web application that integrates several internal and external services combining multiple AI solutions to problems such as multimodal emotion analysis, named-entity recognition and entity linking (normalisation). The communication flow between these components is depicted in figure 38 on the next page. We proceed by discussing their roles and functionalities in a top-down manner.

The User Interface has been built incrementally using React modules starting from a range of components provided open-source by Creative Tim [49] that have been adapted to fit our design requirements. As the user interacts with the UI and performs actions, the Javascript front-end maps them into requests that are sent to a back-end Flask API which carries out the three main functionalities of the application: retrieval of recent posts, classification of the fetched or uploaded posts, and computation of their associated statistics. The Keras models trained over the basic emotion model have been incorporated in the API in order to perform the predictions. These endpoints are presented in table 18 below.

Method	Endpoint	Description
GET	/api/posts/term=?&tag=?&media=?	Fetch posts for the search term and tag.
GET	/api/stats/term=?&tag=?&media=?	Compute statistics for the search term and tag.
POST	/api/predict	Predict emotions for the posts supplied.

Table 18: Flask API endpoints

The posts fetched from the Tumblr API are filtered using Spacy's named-entity recogniser and cached in a non-relational mongoDB database along with their computed statistics, indexed by the search term and its named-entity tag. This allows for faster retrieval in the event of future requests on the same term of interest. However, as the same entity can be referred to in multiple forms, we need to normalise the search term prior to storing it in the database in order to avoid redundancy. For this task, we use the DBPedia Spotlight API which links terms describing one entity to the same canonical identifier. To conclude, this system represents only a single application area of the OceanNet models out of a vast array of opportunities and provides market analysts with a tool for visualising the affective reaction of the public intuitively.

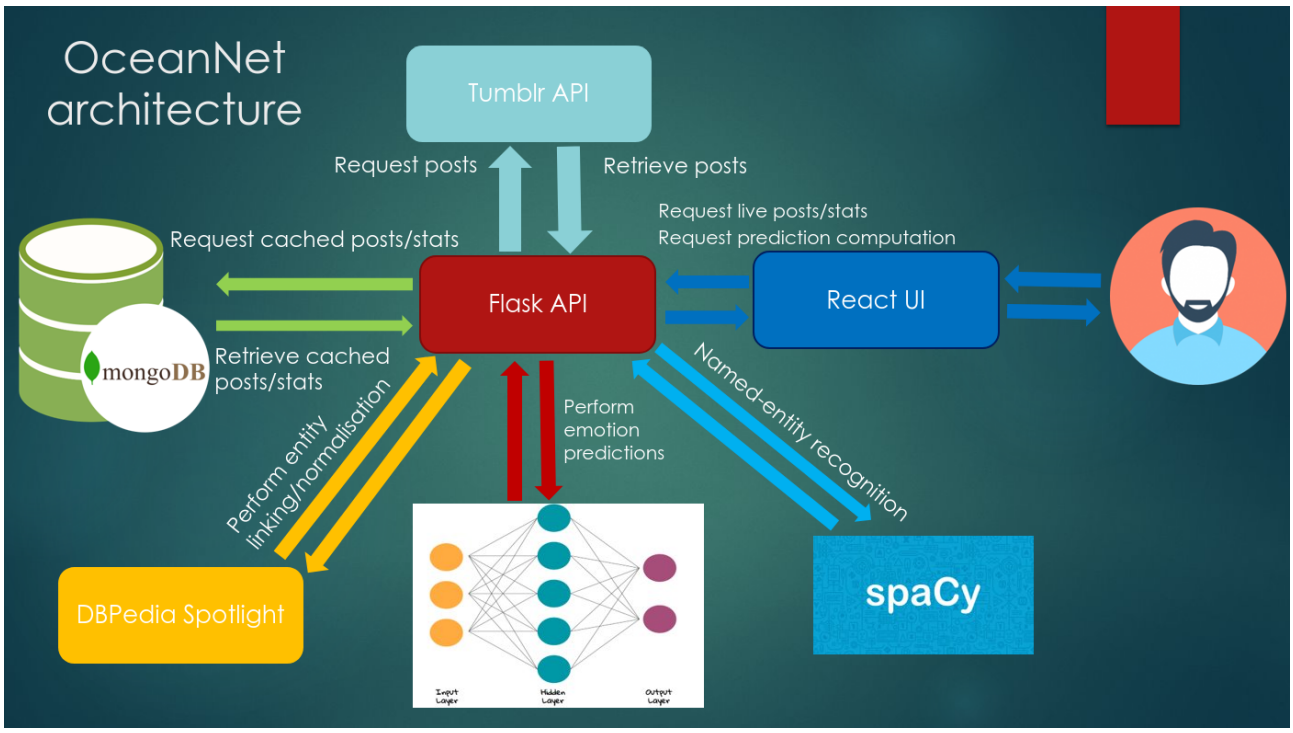

Figure 38: Architecture of the OceanNet visualisation system.

8 Conclusion

In this paper, we discuss the state-of-the-art developments in the fields of multimodal sentiment and emotion analysis and propose a Deep Learning approach to the problems of single- and multi-modal emotion recognition. We leverage the massive amounts of information available on social media as we automatically collect two corpora comprising image and text posts from Tumblr and Twitter, and use them for training our networks in order explore the appositeness of these platforms for the emotion recognition tasks. Finally, the evaluation of the proposed neural architectures is performed over the basic and DeepSentiment's emotion spectra after carefully tuning the hyperparameters.

The top-performing topology for the textual emotion recognition network has been identified throughout a series of grid searches over the number of LSTM modules, neurons per layer and drop-out rates. The final architecture comprises a look-up table of word embeddings followed by a single LSTM layer with 1024 neurons and 20% dropout. Further, class balancing methods have been explored in order to address the uneven distribution of samples across emotions in our datasets. We concluded that SMOTE provides the best overall and per-class performance compared to the weighted loss and random over-sampling techniques. However, the rare affects in DeepSentiment's emotion model are generally mistaken for popular emotions as the network is slightly biased towards the frequent classes. This motivates the need to collect larger and more balanced corpora of multimodal posts as we observe this pattern throughout all analysis modalities, most prominently in the visual recognition task.

The transferability of affective knowledge across Twitter and Tumblr has been researched in order to determine the ability of a single model to predict affects from multiple platforms. We discover that while one model may capture affective representations shared across several social networks, the vast amount of variation in forms of expression from medium to medium limits the ability of a single-input neural network to generalise over the entire social media.

The emotion model assembled by DeepSentiment captures complex affects required in their comprehensive study of emotions on social media. Training our networks on this spectrum of affects provides us with a frame of comparison between the performance achieved by our models and DeepSentiment's. We report the same accuracy as DeepSentiment for text-only emotion recognition, while our transfer learning approach on InceptionV3 outperforms by over 5% their accuracy on the visual emotion analysis task. Finally, the accuracy of our multi-input network that processes both textual descriptions and images is 10% below that of DeepSentiment. We argue that the artificial nature of the more recent posts in our Tumblr dataset accounts for this drop in performance.

The basic model of six emotions has been popular in literature due to its suitability to a variety of application areas. The attention of our models towards all classes has improved on this narrower spectrum as they now seem to grasp more precisely the underlying structure of emotions and learn their characteristic features. We report 79% and 62% accuracy scores obtained on the text-only and image-only prediction tasks, respectively, using the Tumblr dataset; furthermore, taking into account both modalities renders a score of 75%. The drop in performance caused by looking at the visual component alongside the text suggests that images do not necessarily complement or emphasize the description, but merely confuse the models during training. Finally, we report state-of-the-art results when training our LSTM network on the Twitter dataset as the network achieves 97% accuracy and has learned to correctly predict the entire spectrum of affects. This massive increase is fueled by the clarity and conciseness of tweets when compared to their more poetic Tumblr counterparts. Further improvements can focus on the integration of BERT in the text module as well as the collection of larger corpora of social media posts in order to create a universal framework of models that solve the emotion recognition problem across all social media platforms.

To conclude, we encapsulated our state-of-the-art research on the Deep Learning models addressing the problems of textual, visual, and multimodal emotion recognition together with a visualisation tool for market analytics platforms. This product displays the affective evolution and emotion distribution statistics from individual predictions of Tumblr posts retrieved in real-time using a variety of intuitive graphs and charts. This application area of emotion analysis helps businesses, governments, and organisation understand the perception of their audiences and provides them with actionable insights. Furthermore, mental health services and current research in human-computer interaction represent other fields that could also benefit from the performance provided by our neural networks.

9 Appendix

9.1 Emotion synonyms used in the dataset collection

The near-synonyms used in the dataset collection process:

- **happy:** happines, happily, happier, joy, joyful
- **calm:** -
- **sad:** unhappy, sorrow
- **scared:** afraid, scare, scaring, fear, frightened, terror
- **bored:** boring
- **angry:** anger
- **annoyed:** annoy, annoying
- **love:** loved, loving, affection
- **excited:** excitement, enthusiastic, enthusiasm
- **surprised:** surprise
- **optimistic:** optimism, hope, hopeful
- **amazed:** amaze, amazing
- **ashamed:** -
- **disgusted:** disgust
- **pensive:** contemplating, contemplative, contemplate

References

[1] Vasileios Hatzivassiloglou, Kathleen R. McKeown. *Predicting the Semantic Orientation of Adjectives.* ACL & EACL 1997

[2] https://www.aclweb.org/anthology/P02-1053/ *Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews* ACL 2002

[3] Alexander Pak, Patrick Paroubek. *Thumbs up? Sentiment Classification using Machine Learning Techniques.* EMNLP 2002

[4] Jonathan Read. *Using emoticons to reduce dependency in machine learning techniques for sentiment classification.* ACLstudent '05: Proceedings of the ACL Student Research Workshop

[5] Alec Go, Richa Bhayani, Lei Huang. *Twitter Sentiment Classification using Distant Supervision.* Final Projects from CS224N forSpring 2008/2009 at The Stanford Natural LanguageProcessing Group

[6] Alexander Pak, Patrick Paroubek. *Twitter as a Corpus for Sentiment Analysis and Opinion Mining.* LREC 2010

[7] Mehrabian, A., Ferris, S.R. *Inference of attitudes from nonverbal communication in two channels.*. J. Consult. Psychol. 31(3), 248 (1967)

[8] Anthony Hu, Seth Flaxman. *Multimodal Sentiment Analysis To Explore the Structure of Emotions.* KDD 2018

[9] Daniel Todd Gilbert. Stumbling On Happiness. A.A. Knopf, New York, 2006.

[10] Rie Johnson, Tong Zhang *Effective Use of Word Order for Text Categorization with Convolutional Neural Networks.* NAACL HLT, 2015

[11] Rie Johnson, Tong Zhang *Semi-supervised Convolutional Neural Networks for Text Categorization via Region Embedding.* NIPS, 2015

[12] Jeffrey Pennington, Richard Socher, Christopher Manning. *Glove: Global Vectors for Word Representation.* EMNLP, 2014

[13] D. Borth, R. Ji, T. Chen, T. Breuel and S.-F. Chang. *Large-scale visual sentiment ontology and detectors using adjective noun pairs.* ACM International Conference on Multimedia (2013)

[14] Mohammad Soleymani, David Garcia, Brendan Jou, Björn Schuller, Shih-Fu Chang, Maja Pantic. *A survey of multimodal sentiment analysis.* Image and Vision Computing, Volume 65, September 2017

[15] Amir Zadeh, Rowan Zellers, Eli Pincus, Louis-Philippe Morency. *MOSI: Multimodal Corpus of Sentiment Intensity and Subjectivity Analysis in Online Opinion Videos.* IEEE Intelligent Systems 31.6 (2016): 82-88

[16] Soujanya Poria, Erik Cambria, Devamanyu Hazarika, Navonil Majumder, Amir Zadeh, Louis-Philippe Morency. *Context-Dependent Sentiment Analysis in User-Generated Videos.* Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2017

[17] Brendan Jou, Tao Chen, Nikolaos Pappas, Miriam Redi, Mercan Topkara, Shih-Fu Chang. *Visual Affect Around the World: A Large-scale Multilingual Visual Sentiment Ontology.* Proceedings of the 23rd ACM international conference on Multimedia, 2015

[18] Jonathan Posner, James A. Russell, and Bradley S. Peterson. *The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology.* Development and psychopathology, Volume 17, Issue 3, 2005

[19] Benedek Kurdi, Shayn Lozano, and Mahzarin R Banaji. *Introducing the Open Affective Standardized Image Set (OASIS).* Behavior Research Methods 49(2), 2017.

[20] Galen Andrew, Raman Arora, Jeff Bilmes, Karen Livescu. *Deep Canonical Correlation Analysis.* Proceedings of the 30th International Conference on Machine Learning, PMLR 28(3):1247-1255, 2013.

[21] Wei Liu, Jie-Lin Qiu, Wei-Long Zheng, Bao-Liang Lu. Multimodal Emotion Recognition Using Deep Canonical Correlation Analysis. CVPR, 2019

[22] Jie-Lin Qiu, Wei Liu, Bao-Liang Lu. *Multi-view Emotion Recognition Using Deep Canonical Correlation Analysis.* 25th International Conference, ICONIP 2018, Siem Reap, Cambodia, December 13–16, 2018, Proceedings, Part V

[23] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean. *Efficient Estimation of Word Representations in Vector Space.* ICLR, 2013

[24] Jeffrey Pennington, Richard Socher, Christopher Manning. *Glove: Global Vectors for Word Representation.* EMNLP, 2014

[25] Scott Deerwester , Susan T. Dumais , George W. Furnas , Thomas K. Landauer , Richard Harshman. *Indexing by latent semantic analysis.* Journal of the American Society for Information Science, Volume 41, Issue 6, 391-407, 1990

[26] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer. *Deep contextualized word representations.* NAACL, 2018

[27] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* NAACL, 2019

[28] Sreeja P.S., G.S. Mahalakshmi. *Emotion Models: A Review.* International Journal of Control Theory and Applications 10(8):651-657, 2017

[29] Paul Ekman. *Pan-Cultural Element in Facial Displays of Emotions.* SCIENCE, Volume 164, 86-88, 4 April 1969

[30] David Matsumoto. *Paul Ekman and the legacy of universals.* Journal of Research in Personality 38(1):45-51, February 2004

[31] Carlo Strapparava, Rada Mihalcea. *Learning to Identify Emotions in Text.* Proceedings of the 2008 ACM Symposium on Applied Computing (SAC), Fortaleza, Ceara, Brazil, March 16-20, 2008

[32] David Watson, Lee Anna Clark. *The PANAS-X: Manual for the Positive and Negative Affect Schedule - Expanded Form.*

[33] Robert Plutchik. *The Nature of Emotions.* American Scientist 89 (2001), 344.

[34] Yann LeCun , Yoshua Bengio Geoffrey Hinton. *Deep learning.* NATURE — VOL 521 — 28 MAY 2015

[35] Yoav Goldberg. *A Primer on Neural Network Models for Natural Language Processing.* Journal of Artificial Intelligence Research, September 2016

[36] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, Zbigniew Wojna. *Rethinking the Inception Architecture for Computer Vision.* Computer Vision and Pattern Recognition, 2016

[37] John Hewwit. *Finding Syntax with Structural Probes.* Natural Language Processing Research, 2019

[38] Dallas Geerlings. *Twitter Engagement Study: Photo vs. Text Tweets.* brandnetworks

[39] *How to Increase Twitter Engagement by 324%.* quicksprout

[40] Mahendran Venkatachalam. *Recurrent Neural Networks.* Medium, 2019.

[41] Michael Nguyen. *Illustrated Guide to LSTM's and GRU's: A step by step explanation.* Medium, 2018.

[42] Sik-Ho Tsang. *Review: Inception-v3 — 1st Runner Up (Image Classification) in ILSVRC 2015.* Medium, 2018

[43] German Lahera. *Unbalanced Datasets What To Do About Them.* Medium, 2019.

[44] Rafael Alencar. *Resampling strategies for imbalanced datasets.* Kagle, 2018.

[45] Tumblr user: lapfullofcatfur. Happy post.

[46] Tumblr user: girlwithlandscape. Surprise post.

[47] Twitter user: YorkshireSheperdess. Calm post.

[48] Twitter user: Kevin Nevis. Ashamed post.

[49] Creative Tim.