Is Deep Learning a Valid Approach for Inferring Subjective Self-Disclosure in Human-Robot Interactions?

Henry Powell
Institute of Neuroscience and Psychology,
University of Glasgow,
Glasgow, UK
h.powell.2@student.gla.ac.uk

Guy Laban
Institute of Neuroscience and Psychology,
University of Glasgow,
Glasgow, UK
Guy.Laban@Glasgow.ac.uk

Jean-Nöel George
School of Psychology,
U*niversity of Glasgow*
Glasgow, UK

Emily S. Cross
Institute of Neuroscience and Psychology,
University of Glasgow,
Glasgow, UK
Department of Cognitive Science,
Macquarie University
Sydney, New South Wales, Australia
emily.cross@glasgow.ac.uk

***Abstract*—One limitation of social robots has been the ability of the models they operate on to infer meaningful social information about people's subjective perceptions, specifically from non-invasive behavioral cues. Accordingly, our paper aims to demonstrate how different deep learning architectures trained on data from human-robot, human-human, and human-agent interactions can help artificial agents to extract meaning, in terms of people's subjective perceptions, in speech-based interactions. Here we focus on identifying people's perceptions of their subjective self-disclosure (i.e., to what extent one perceives to be sharing personal information with an agent). We approached this problem in a data-first manner, prioritizing high quality data over complex model architectures. In this context, we aimed to examine the extent to which relatively simple deep neural networks could extract non-lexical features related to this kind of subjective self perception. We show that five standard neural network architectures and one novel architecture, which we call a Hopfield Convolutional Neural Network, are all able to extract meaningful features from speech data relating to subjective self-disclosure.**

***Index Terms*—Datasets, Neural Networks, Speech Recognition, Human-robot Interaction, Behavioral Health, Non-intrusive sensing technology, Communication, Perception, Affective computing**

I. Introduction

Social robots do not (yet) offer the same opportunities as humans for social interactions (see (1)). In particular, social robots are limited in their ability to infer meaningful social information from speech disclosures (2). Most humans, on the other hand, effortlessly engage in theory of mind (i.e., inferring a person's mental state) and are generally capable of using these abilities when communicating through speech (3; 4; 5). While humans can intuitively infer complex social information regarding a conversation partner, artificial agents need to synthesize and analyze multiple kinds (or channels) of data from a human interaction partner in order to appropriately and accurately "read" complex social meanings (6). One important facet of an interaction that human's are adept at detecting is self-disclosure, a complex social dynamic that consists of multiple dimensions. In this study we are particularly interested in *subjective* self-disclosure, i.e. the amount of personal information one perceives to be sharing during an interaction (7; 8; 8; 9). The aim of the current study is to experimentally validate the efficacy of standard deep learning models for classifying a person's subjective self-reported levels of self-disclosure in human-robot, human-human, and human-agent interactions.

We approach this problem in a data-centric manner, i.e., by applying robust experimental design methodology and prioritising the collection of high quality data over complex computational models. The data collection procedures for the raw data used in the present study allows for maximal control over the robustness and quality of the data collected. Further, allowing participants to rate their own interactions carried the benefit that each sample was accurately labelled. To probe this problem, we investigated the effectiveness of two different standard feature sets from the speech recognition literature: log-mel and eGeMAP features. Our experiments

The authors gratefully acknowledge funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement 677270 to E.S.C.), the Leverhulme Trust (PLP-2018-152 to E.S.C.), and the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie to ENTWINE, the European Training Network on Informal Care (grant agreement no. 814072 to E.S.C.)

978-1-6654-0731-1/22/$31.00 ©2022 IEEE

were conducted using 5 standard neural network architectures as well as a novel architecture that replaced the LSTM layer in a CNNLSTM with a Hopfield layer.

II. Data Set and Data Collection

To generate data for the models, three laboratory experiments were conducted, as reported in detail previously (10; 11). The three laboratory experiments ($N1 = 26; N2 = 27; N3 = 61$) consisted of within-subjects experimental designs with three treatments. In a randomized order, participants were asked one (in the first experiment) or two (in the second and third experiments) pre-defined questions about their everyday life experiences by each of the three agents: (1) a humanoid social robot (NAO by Softbank Robotics), (2) a human, or (3) a disembodied agent (a "Google mini" voice assistant) The agents communicated the same pre-scripted questions via the Wizard of Oz (WoZ) technique controlled by the experimenter, demonstrating different cues that corresponded appropriately to their embodiment. The questions' topics were randomly allocated to the agents, and the questions within each topic were randomly ordered. All three experiments took place in a sound-isolated recording laboratory. The recording room was completely soundproofed to ensure the highest possible sound quality for the recordings to facilitate offline analyses. After each interaction, participants answered a questionnaire reporting their level of perceived self-disclosure to the agent via 10 items of an adapted version of Jourard's Self-Disclosure Questionnaire (12).

III. Feature Sets and Data Augmentation

We were interested in examining the effects that two different kinds of feature sets would have on the deep learning models that we used in our experiments. The first feature set chosen was log-mel spetrograms and their cepstral coefficients. This data representation was chosen because representing inputs in log-mel space has been shown to be an effective data representation for complex speech recognition tasks (13; 14). Log-mel spectra are two-dimensional representations of one-dimensional amplitude signals that are produced by first applying a fast-Fourier transform to the signal using a sliding window. The Fourier transformed windows, which are now in 2D, are then concatenated to produce a time-series of amplitude spectra in the Hz domain. To produce mel-spectra, these time series are then transformed from the Hz domain into the mel-frequency domain, a log-scale domain which matches the way in which humans perceive the distances between two pitches. We used the following standard equation to convert a frequency f to a mel-frequency m:

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{1}$$

The cepstral coefficients are produced by taking a cosine-transform of the logs of the powers of the individual mel frequencies. To produce a singular feature set we then concatenated the log-mel spectra with their associated cepstral coefficients. For our experiments we computed 128 mel-filter banks and applied them to the Fourier windows and then computed 20 cepstral coefficients resulting in a 148 dimensional feature space for our input data.

The second feature set we chose to investigate were so called "hand crafted" features from the Geneva Minimalistic Acoustic Parameter Set (eGeMAPS)(15). This is an acoustic feature set designed to avoid over fitting in machine learning models by not overwhelming the models with thousands of brute-forced features. eGeMAPS contains 88 statically computed low-level descriptors of an audio signal including frequency, amplitude, and spectral parameters. To create a time series of each WAV data point we took eGeMAP features of sliding windows of the amplitude data in 10ms segments.

The raw data from (11; 16) consisted of 625 interactions as waveform audio files. The authors found that no participants' average perceived self disclosure score reached the score of 7 on the scale, so this class was removed shifting the scale to 1-6. There was also a large degree of bias toward the central scores in the scale meaning that a majority of participants scored their interactions in the range $[2, 5]$ creating a large degree of class imbalance. This is particularly problematic as the most underrepresented classes were the scores of 1 and 6. To combat this class imbalance problem and produce a more balanced dataset we augmented the raw data using two feature sets that we considered for our experiments: log-mel features and eGeMAPS.

A. Log-Mel Features

The first augmentation technique applied to the log-mel version of the data was vocal-tract length perturbation (17). The length of a person's vocal tract is one of the key factors in determining the qualities of that person's voice. The intuition behind vocal tract length perturbation is that if we can computationally mimic a shift in the length of the performer's vocal tract by transforming the data then we will have a new example of that data point because it simulates the speech segmented being uttered by a different person. This changes the quality of the voice in the data point without changing the underlying features of the data that we are aiming to capture in the model. Visually this has the effect of stretching the mel-spectrogram slightly in the frequency domain and is similar to image warping techniques used in image classification tasks. We computed vocal tract length perturbation by shifting the central frequency of the mel filter banks used to transform the data from the Hz domain to the log-mel domain using a fixed warping coefficient and the following formula:

$$f' = \begin{cases} f\alpha & f \leq F_{hi}\frac{min(\alpha,1)}{\alpha} \\ S/2 - \frac{S/2 - F_{hi} min(\alpha,1)}{S/2 - F_{hi}\frac{min(\alpha,1)}{\alpha}}(S/2 - f) & otherwise \end{cases} \tag{2}$$

Where f refers to the starting frequency, f' is the transformed frequency, α is the fixed warping coefficient, and F_{hi} is a boundary frequency chosen to cover the significant formants in the signal. As in (17) we set $Fhi = 4800$. While drawing the warping coefficient from a uniform distribution in a certain range is a common technique (17; 18) we found, in line with (19), that choosing fixed warping coefficients of 0.9 and 1.1 produced the best results. This also allowed us to apply two separate perturbations to the data in the underrepresented classes.

B. eGeMAP features

Since the eGeMAP time series we produced from the original raw audio files do not naturally lend themselves to the same techniques for augmentation detailed above we instead used weighted random sampling to ensure that the network was being trained on an even number of examples from each class. Weighted random sampling, a development of sequential or uniform random sampling (20), assigns a weight to each example in a training dataset where the weight is the reciprocal of the probability that example would be chosen at random. This means that examples from underrepresented classes are more likely to be chosen in a batch of input data that is used to train a deep learning model. As a result the model is shown more examples from under represented classes. We found that weighted random sampling in this way increased the stability of the learning procedure of our models when trained on eGeMAP features despite recent work that has shown that under specific assumptions about network architecture and learning algorithms, importance related sampling can have a limited positive effect on network training (21). eGeMAP features were extracted from the raw WAV files using the opensmile toolkit in python (22).

IV. Deep Learning Experiments

In our experiments we used five standard deep neural network architectures and one novel architecture that we designed to leverage the spatio-temporal nature of the input data space, as well as make use of some key advances in time series modelling in the field of artificial neural networks over the past couple of years.

A. Neural Network Architectures

1) Linear Neural Network: Our linear network consisted of five fully connected layers where each hidden layer consisted of 1024 neurons. We applied drop-out and batch normalization to each layer to prevent over-fitting. Each layer was then passed through an ReLU non-linear activation function before its output was passed to the next layer. The output layer consisted of a single neuron so as to implement a regression problem. The architecture of this stack of linear layers was also used as the classification stack in each one of the other networks that we used in our experiments.

2) Convolutional Neural Network: Convolutional neural networks (23) have been used successfully in a number of tasks related to time series modelling (24). To test the efficacy of these architectures we constructed a network with two one-dimensional convolutional layers and a linear stack for classification. The first convolutional layer passes a nx5 convolutional kernel with a stride of 1 over each data sample along the time dimension where n is the number of features for each problem. The number of feature maps produced by this first layer was $\frac{t}{5}$ where t is the number of time steps in each sample fed to the network. This produced 35 feature maps for the log-mel feature set and 15 for the eGeMAP feature set. Each of these feature maps was then fed through an ReLU non-linearity before being summarised by a 1D max pooling layer with a 3x3 kernel. The second convolutional layer contained 15 nx5 kernels with a stride of 1 and a max pooling layer with the same parameters as in the previous layer. Both layers also contained 1D batch normalisation to prevent overfitting. Finally the output of the second convolutional layer was fed to a linear classification stack that mirrors the structure of the linear neural network above.

3) Long Short-Term Memory Network: Long Short-Term Memory (LSTM) networks have been shown to produce state-of-the-art results on a number of time series problems including a number of audio classification tasks from emotion recognition (25). For our experiments we used a simple single layer LSTM network with 296 LSTM cells. The output of this layer was then fed to a linear classification stack as above.

4) Convolutional Long Short-Term Memory Network: Convolutional Long Short-Term Memory Networks (CNNLSTM)(26) utilize a hybrid-architecture methodology whereby an input data point, usually a multi-variate time series, is fed through m either one-dimensional or two-dimensional convolutional layers supplemented with max pooling for averaging the features learned by the convolutional kernels and dropout for regularization. These feature maps are then fed through n long short-term memory layers to extract temporal features. Finally the data is fed through p fully connect linear layers and a softmax layer for classification. Our version of a CNNLSTM simply combines the three architectures above: The input is fed into a two-layer 1D convolutional stack then into a single LSTM layer before being fed into a linear classifier.

5) Hopfield Network: The limitation of CNNLSTM models is that their capacity to store temporally extended relations between points in data are limited by the LSTM layers. While LSTMs are partial solutions to the exploding/vanishing gradient (27) problems they still suffer from poor performance when faced with longer sequences. More recently attention based models were introduced for natural language processing tasks (28) that improved on the base performance of LSTMs (and recurrent architectures more generally) by allowing the model to learn a vector embeddingthat teaches the model which parts of a sentence are relevant to which other parts. Attention based LSTM models have proven successful in a number of natural language processing tasks such as sentiment classification (29; 30) and emotion recognition (31).

To test this we created a Hopfield network that simply replaced the LSTM layer in our LSTM model with a Hopfield layer. Since the Hopfield layer cannot encode temporal information from the data natively (as is done via the existence of recurrent connections between neurons in the hidden layers of recurrent neural networks such as LSTMs) we use positional encoding as in (32). The positional encoding is a static matrix generated using the following formula:

$$\begin{aligned} PE_{(pos,2i)} &= sin(pos/10000^{2i/dmodel}) \\ PE_{(pos,2i+1)} &= cos(pos/10000^{2i/dmodel}) \end{aligned} \tag{3}$$

where pos is the position, i is the dimension of the input (in our case this refers to the 128 filter bands from the mel-spectrograms or the 88 eGeMAP features), and d is the dimension of the model. The reason for using a static encoding over a learned encoding via embedding was two-fold. Firstly, testing showed that using a learned embedding had negative effects on the model's performance, and secondly as is noted in (32) the static positional encoding has the benefit that it generalizes to input lengths greater than that which the model was trained on.

6) Hopfield Convolutional Neural Network: For the final model we designed a network architecture that combined the spatio-temporal representational power of hybrid networks like CNNLSTMs while aiming to improve on their performance by taking inspiration from the developments in the attention field Our model, which we call a Hopfield Convolutional Neural Network, replaces the LSTM layers in traditional CNNLSTMs with a Hopfield layer. In this model we again simply replace the LSTM layer in our CNNLSTM architecture with a Hopfield layer and use a static positional encoding (as in (32)) to inform the Hopfield layer about the temporal position of each observation in each data point.

B. Experiments

We split the data into training and testing datasets. The testing set in each case contained participants that had not been seen by the model in the training phase so as to reflect the kinds of examples that it might see in a real world scenario. Testing participants were selected such that the testing set contained as even a balance of examples from the classes in each problem as possible and that the number of testing to training samples that the model experienced during training was between 10% and 20%. The reason the train-testing split was inconsistent was due to the fact that we split the training and testing sets by participant. Each participant represented two or three interactions, regularly with different scores and lengths of time. Therefore one participant might represent significantly more samples per class than another participant when the interactions were split into windows of a fixed length. Finally to ensure consistency in our comparison between models and between feature sets, the same training and testing participants were used in each case.

We split the input data up into windows of constant length: 150 frames of data for log-mel features and 75 frames for eGeMAP features as these were found to produce the best results for each problem. Each network was trained on mini batches of 200 samples (i.e. 200 windows of a given length) from the training data set over a period of 300 epochs for the log-mel feature set and 100 epochs for the eGeMAP feature set. The differences in the epoch hyper-parameters were due to the speeds at which the networks tended to converge in each case. For each network we used the ADAM optimiser (33) and a negative mean-squared error lost function[1]. We trained the architectures on the log-mel and eGeMAP feature sets separately to explore how effective each of these literature-standard feature types were at capturing informative features from the data for this classification problem. Each model was validated according to an accuracy metric defined as the percentage of correctly classified samples from a testing set i.e. what percentage of examples from the testing set the model correctly identified as belonging to a ground-truth self-disclosure score. Since we were dealing with a regression problem we computed the classification accuracy by rounding the regression score for each input to the nearest integer. We then compared this result to the ground truth integer score when computing the accuracy of the input batch.

V. Results

The results of our experiments are displayed in table I. We found that all networks for both the mel-spectrogram and eGeMAP feature sets learned meaningful features from the data such that they were able to achieve accuracy scores significantly above chance. On the mel-spectrogram features we found that all models performed effectively identically scoring around 48% in each case, while for the eGeMAP features the linear net was the most accurate with a score of 43.52%. We further found that log-mel features were the most informative, leading to significantly better accuracy scores than the eGeMAP features. For both the mel-spectrogram and the eGeMAP features we found that the networks tended to overfit the training data. To combat this we set the network dropout values to 10% for the mel-spectrogram features and 90% for the eGeMAP features to account for the degree of over fitting that we experienced in both cases. Nonetheless, we found that learning in all networks was difficult as is clear from the results. We hypothesize that the failure to achieve much higher accuracy scores may be able to be put down to the difficulty of the task. It's intuitive that an important way in which we ascertain whether someone is disclosing personal information is informed in no small part by the lexical properties of their speech i.e. what it is they are saying as opposed to how they are saying it. Since lexical features were absent from the feature sets (since we were specifically interested in investigating whether networks could learn non-lexical properties of speech) it makes sense that the task would be significantly harder than if we had included lexical based features. However it is clear from the results that the data were informative enough to allow the networks to lean non-lexical features despite the intuitively challenging nature of the task.

VI. Discussion and Conclusions

This study provides novel scientific and technical contributions to the HRI and affective computing research communities in a number of ways. To our knowledge, this is the first

[1] A table containing all network hyperparameters for both feature sets is displayed in II

TABLE I
SELF-DISCLOSURE MODEL ACCURACY FOR MEL-SPECTROGRAM AND eGeMAPS FEATURE SETS

Model Type	Mel-Spec Accuracy(%)	eGeMAPS Accuracy(%)
Chance	16.67	16.67
LNN	48.2	43.52
CNN	48.28	42.42
LSTM	48.34	41.05
CNNLSTM	48.13	40.08
Hopfield	47.8	42.74
HopfieldCNN	48.28	42.85

TABLE II
NETWORK HYPERPARAMETERS

Hyperparameter	eGeMAP Models (%)	MelSpec Models (%)
Learning Rate	0.1	0.1
Epochs	100	300
Input Size (frames)	75	150
Batch Size	200	200
Dropout	0.9	0.1
Loss Function	MSE	MSE

attempt to investigate deep learning's ability to extract features related to a person's subjective experience from their speech. By using genuine data that was collected in HRIs we will be able to extend these and implement those insights to further understand how humans communicate with robots, and by applying these sort of models we will be further ahead in granting robots the ability to understand humans subjective perceptions. While the presented architectures are relatively straight-forward, these models (and other similar models that use the same approach) conceptually mark small steps towards creating robots that understand people from their subjective point of view by synthesizing available non-intrusive behavioral cues. Adapting the architectures presented here could help equip artificial agents to understand humans better. Nevertheless, our results however do show that much improvement can and should be made before deep learning platforms are seriously considered for being introduced or implemented into social robots. We do, however, believe that our results show that such progress is possible and that there are promising avenues for future research in this space.

REFERENCES

[1] E. S. Cross, R. Hortensius, and A. Wykowska, "From social brains to social robots: applying neurocognitive insights to human-robot interaction," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 374, no. 1771, p. 20180024, 2019.

[2] A. Henschel, G. Laban, and E. S. Cross, "What Makes a Robot Social? A Review of Social Robots from Science Fiction to a Home or Hospital Near You," *Current Robotics Reports*, no. 2, pp. 9–19, 2021.

[3] C. Catmur, E. S. Cross, and H. Over, "Understanding self and others: from origins to disorders," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 371, no. 1686, p. 20150066, 2016.

[4] D. Premack and G. Woodruff, "Does the chimpanzee have a theory of mind?," *Behavioral and Brain Sciences*, vol. 1, no. 4, pp. 515–526, 1978.

[5] L. J. Byom and B. Mutlu, "Theory of mind: mechanisms, methods, and new directions," *Frontiers in human neuroscience*, vol. 7, p. 413, aug 2013.

[6] A. Kappas, R. Stower, and E. J. Vanman, "Communicating with robots: What we do wrong and what we do right in artificial social intelligence, and what we need to do better," 2020.

[7] C. Antaki, R. Barnes, and I. Leudar, "Diagnostic formulations in psychotherapy," *Discourse Studies*, vol. 7, no. 6, pp. 627–647, 2005.

[8] H. Kreiner and Y. Levi-Belz, "Self-Disclosure Here and Now: Combining Retrospective Perceived Assessment With Dynamic Behavioral Measures," *Frontiers in Psychology*, vol. 10, p. 558, 2019.

[9] J. Omarzu, "A Disclosure Decision Model: Determining How and When Individuals Will Self-Disclose," *Pers Soc Psychol Rev*, vol. 4, no. 2, pp. 174–185, 2000.

[10] G. Laban, J.-N. George, V. Morrison, and E. S. Cross, "Tell me more! assessing interactions with social robots from speech," *Paladyn, Journal of Behavioral Robotics*, vol. 12, no. 1, pp. 136–159, 2021.

[11] G. Laban, V. Morrison, and E. S. Cross, "Let's talk about it! subjective and objective disclosures to social robots," p. 328–330, Association for Computing Machinery, 2020.

[12] S. M. Jourard, *Self-disclosure: An experimental analysis of the transparent self.* Oxford, England: John Wiley, 1971.

[13] H. Meng, T. Yan, F. Yuan, and H. Wei, "Speech emotion recognition from 3d log-mel spectrograms with deep learning network," *IEEE Access*, vol. 7, pp. 125868–125881, 2019.

[14] C. Etienne, G. Fidanza, A. Petrovskii, L. Devillers, and B. Schmauch, "Speech emotion recognition with data augmentation and layer-wise learning rate adjustment," *CoRR*, vol. abs/1802.05630, 2018.

[15] F. Eyben, K. R. Scherer, B. W. Schuller, J. Sundberg, E. André, C. Busso, L. Y. Devillers, J. Epps, P. Laukka, S. S. Narayanan, and K. P. Truong, "The geneva minimalistic acoustic parameter set (gemaps) for voice research and affective computing," *IEEE Transactions on Affective Computing*, vol. 7, no. 2, pp. 190–202, 2016.

[16] "Anonymized for peer-review process - anonymized version of the paper available upon request,"

[17] N. Jaitly and E. Hinton, "Vocal tract length perturbation (vtlp) improves speech recognition," 2013.

[18] C. Kim, M. Shin, A. Garg, and D. Gowda, "Improved vocal tract length perturbation for a state-of-the-art end-to-end speech recognition system," pp. 739–743, 09 2019.

[19] I. Rebai, Y. BenAyed, W. Mahdi, and J.-P. Lorré, "Improving speech recognition using data augmentation and acoustic model fusion," *Procedia Computer Science*, vol. 112, pp. 316 – 322, 2017. Knowledge-Based and Intelligent Information Engineering Systems: Proceedings of the 21st International Conference, KES-20176-8 September 2017, Marseille, France.

[20] J. H. Ahrens and U. Dieter, "Sequential random sampling," *ACM Trans. Math. Softw.*, vol. 11, p. 157–169, June 1985.

[21] J. Byrd and Z. C. Lipton, "Weighted risk minimization & deep learning," *CoRR*, vol. abs/1812.03372, 2018.

[22] F. Eyben, M. Wöllmer, and B. Schuller, "Opensmile: The munich versatile and fast open-source audio feature extractor," in *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, (New York, NY, USA), p. 1459–1462, Association for Computing Machinery, 2010.

[23] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

[24] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," *CoRR*, vol. abs/1404.2188, 2014.

[25] M. Schmitt and B. Schuller, "Deep recurrent neural networks for emotion recognition in speech," in *Fortschritte der Akustik - DAGA 2018: Proceedings der 44. Jahrestagung für Akustik, München, Deutschland, 19-22 März 2018* (B. Seeber, ed.), 2018.

[26] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4580–4584, 2015.

[27] R. Pascanu, T. Mikolov, and Y. Bengio, "Understanding the exploding gradient problem," *CoRR*, vol. abs/1211.5063, 2012.

[28] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014. cite arxiv:1409.0473Comment: Accepted at ICLR 2015 as oral presentation.

[29] Y. Wang, M. Huang, X. Zhu, and L. Zhao, "Attention-based lstm for aspect-level sentiment classification," in *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 606–615, 2016.

[30] M. Yang, W. Tu, J. Wang, F. Xu, and X. Chen, "Attention-based lstm for target-dependent sentiment classification," in *Proceedings of the thirty-first AAAI conference on artificial intelligence*, pp. 5013–5014, 2017.

[31] Y. Xie, R. Liang, Z. Liang, C. Huang, C. Zou, and B. Schuller, “Speech emotion classification using attention-based lstm,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 11, pp. 1675–1685, 2019.

[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems 30* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), pp. 5998–6008, Curran Associates, Inc., 2017.

[33] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *International Conference on Learning Representations*, 12 2014.