HLM12ERC, a Multimodal Model for Emotion Recognition in Conversations: Project Design

Deep Learning on Public Dataset

CM3070, Final Computer Science Project

Hudson Leonardo MENDES
University of London
2023

Page count (except ToC, Images & References): 4 pages

Project Overview

The template chosen for this project was the **Deep Learning on a public dataset**, and the Dataset chosen was the **MELD: A Multimodal Multi-Party Dataset for Emotion Recognition in Conversations**[1].

The present project presents the design for the "**HLM12ERC"** model, a multimodal model for emotion recognition in conversations on the MELD dataset[1], describing the Problem, Domain & Users, Aims & Objectives, Structure of the Project, Evaluation Protocols and Work Plan.

Table of Contents

Problem, Domain & Users

The ability of recognising people's emotions in conversation is not just an essential social skill, but also finds a wide range of commercial applications, such as management & marketing (Strategy development, Brand management, Churn prediction, Preference learning), User interaction (Chatbots, Social Networks), Finance (Investment decision, Economic growth indicator), Politics (Political participation, Public monitoring), Health (Depression treatment, Suicide prevention, Public health forecast, Diagnosis), Education (E-learning)[2].

Aims & Objectives

The present project aims to **introduce a model "HLM12ERC" capable of high accuracy for the emotion classification task against the Test Split of the MELD Dataset[1]**. To pursue that aim, the following objectives are set out:

Objective 1: Create a baseline ERC model inspired in the M2FNet model architecture[3] with "statistical power"[4] against the MELD test set taking the multimodal input from the MELD dataset[1] and predicting an emotion label using and concatenating simple representations of the input and using cross entropy as the objective function;

Objective 2: Evaluate whether the introduction of prompt-engineering to the model, similar to the one used by SPCL-CL-ERC[5] paper, but using the entire dialogue context as prior to producing utterance representations using an auto-regressive language model such as Longformer[6] and MPT-7B-StoryWriter[7] outperforms the simple GloVe-based text representations;

Objective 3: Evaluate whether a model using representations from a State-of-the-Art Face Detection model such as TinaFace(ResNet-50)[8], followed by a pre-trained face embeddings model such as FaceNet[9] to outperforms the baseline model with LeNet-5[12] visual representations;

Objective 4: Evaluate whether using a model using Pre-trained General Purpose Audio Embedding model such as Wave2Vec 2.0[10] outperforms the model with raw wavelength information from audio files;

Objective 5: Evaluate whether a similar fusion network to the one introduced by the M2FNet model[3] outperforms the simple concatenation of features;

Objective 6: Evaluate whether triplet contrastive loss outperforms cross entropy loss;

Objective 7: Compile & Evaluate final model based on best performing components.

Structure of the Project

The following section discusses the candidate (a) Model architectures which will be experimented, (b) the Algorithms that compose this architecture, and the (c) code structure of the project.

Model Architecture

Regardless of the underlying components that will form the final model based on the results of experimentation, due to its multimodal nature, its composition is likely to result in a model *ensemble* of significant complexity.

The architecture is designed to (a) produce embeddings for each one of the modalities, (b) combined these into a single vector representation with the three modalities combined, (c) transformed hierarchically into higher level representations through a feedforward network so they can be finally used to (d) produce a label through a softmax activation, and it can be illustrated as in ***Figure 1***.

Objectives 1 through to 6 will evaluate whether replacing simple baseline representations by richer pre-trained representations, as well as other improvements, are beneficial to the model in terms of accuracy over the MELD test set[1].

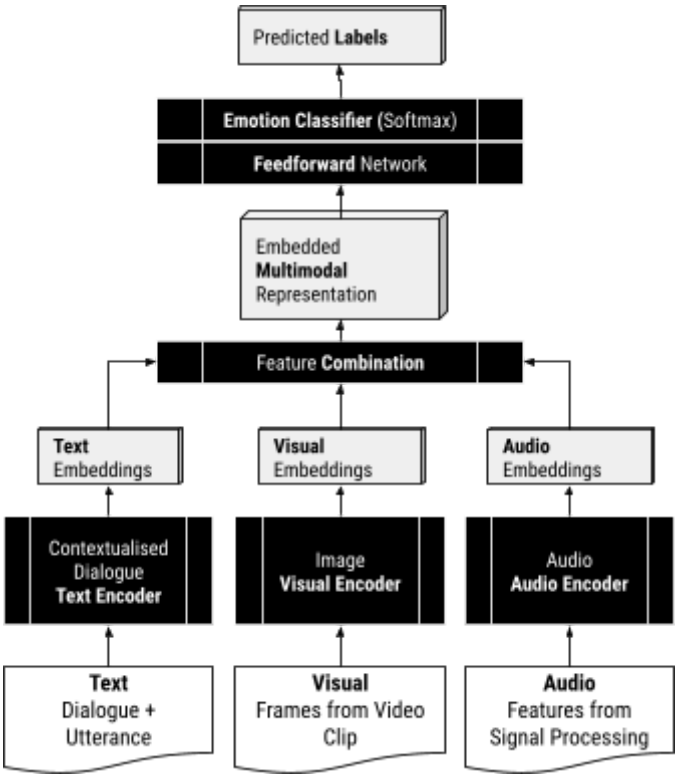

Figure 1: HLM12ERC High-level model architecture.

Algorithms & Equations

Several algorithms will be investigated by the present project and, if they result in significant gains against the MELD test set[1], compose the final model.

Due to their number, complexity and to constraints in the number of pages the present document is limited to, the present section will provide a comprehensive list with a brief description of what these components are, their links to their source research and the reason for which they were chosen.

Baseline Sentence Representation based on GloVe Token Embeddings

GloVe token embeddings that are learned through "global word-word co-occurrence"[11]. Using the mean of the token vectors is a common practice in NLP and it is used as the baseline for the textual representation.

Baseline CNN-based Image Embeddings

Since the publication of the model LeNet-5[12], Feature maps produced by CNNs have become the most fundamental way of generating image representations, rather than using raw pixel data. For

that reason LeNet-5[12] representations are used as baseline for audio embeddings.

Prompt-based Dialog-Contextualised Utterance Text Embeddings with Longformer[6] or MPT-7B-StoryWriter[7]

Both M2FNet and SPCL-CL-ERC use variants of the RoBERTa model which provides a maximum context-window of 512 tokens. M2FNet uses "a variable stack of transformer encoders (...) to learn the inter utterance context"[3] and does not model individual speaker's context, while SPCL-CL-ERC uses prototypical vectors rather than learning inter utterance context in any way, but it models the speaker individual context through prompts[5].

Conversely, the present project attempts to improve the dialogue context modelling by encoding the entire dialogue using a prompt design approach[13] with the purpose of benefiting from the Transformers architecture's well known ability to effectively capture long-term dependencies and context information from its fixed input size[14]. However, a context-window of 512 is not sufficient for that.

Longformer[6] or MPT-7B-StoryWriter[7] are Transformer models that provide a longer context window, 16,000 tokens and 65,000 tokens respectively. With enough computation, the entire dialogue could be used as the context and possibly generate powerful representations. This approach will be evaluated under Objective 2.

TinaFace(ResNet-50)+ FaceNet Visual Embeddings

While the entire video scene of the utterance might have important disambiguation information to which is the right emotion of the speaker, the present work hypothesise that face features carry better signal-to-noise ratio, and will evaluate whether focusing exclusively on faces with TinaFace(ResNet-50)[8], or utilising the richer pre-trained face embeddings FaceNet[9] can improve the model accuracy when compared to simply processing the entire image with the LeNet-5[12] baseline, under Objective 3.

Wave2Vec 2.0 Audio Embeddings

The input data for Wave2Vec 2.0 is the raw waveform of the audio[10], the same input data as the baseline model. However, the baseline model attempts to learn all its representations from the training data. Wave2Vec 2.0, on the other hand, carries forward information from its pre-training phase, which may prove valuable and drive greater accuracy. Hence, Wave2Vec 2.0 will be evaluated as an alternative to Audio Feature Extraction under Objective 4.

Fusion Network as Feature Fusion Mechanism

There are multiple techniques that allow feature fusion, with Mean and Concatenation operations being common approaches. However, meaning for instance combines features irrespective of their importance, while concatenation increases the dimensionality of the hypothesis space being explored. An alternative approach, also explored by M2FNet[3] is allowing the attention mechanism to weigh which features are going to be considered more strongly than others by learning their relationships[14]. A Fusion Network will be investigated as an alternative to multimodal feature fusion.

Triplet Contrastive Learning Objective Function

A triplet loss function based on an anchor, a positive and a negative example can be used to increase distance between contradictory examples and reduce distance between similar examples[9], resulting in better separation of data in the embeddings space. Such an approach is explored by both M2FNet[3] and SPCL-CL-ERC[5] models. The present project will also evaluate whether this can drive an improvement in terms of accuracy.

Libraries & Technologies

The software for the present project will be written in Python, and make extensive use of popular ML python packages, such as the ones described in this section.

PyTorch & HuggingFace for Modelling & Training

HLM12ERC will be modelled and packaged as a PyTorch[15] model. However, instead of writing the training loop manually, the HuggingFace Trainer[16] class shall be used.

Ray Tune for Grid Search & Sklearn for Metrics

The search space defined by the multiple architectural choices that must be made is large enough to require automatic exploration. Grid Search will be implemented for that purpose with Ray[17]. Metrics will be calculated using SKLearn[22] to tell which settings have been the most successful.

Pandas, Librosa, Numpy

The MELD dataset is composed of text, videos and audio, and will be represented in the project as Pandas[18] dataframes. Dataframes, as well as audio read by Librosa[19] is stored in memory as a Numpy[20] array.

Project Organisation

Code-wise, the project will be organised as a (a) jupyter notebook **research.ipynb**, responsible for the MLOps pipeline and (b) a python **package "hlm12erc"** with the model, training and serving code, used by the jupyter notebook, in the following structure:

```
./src/hlm12erc/modelling/
./src/hlm12erc/training/
./src/hlm12erc/serving/
./tests/
./setup.py
./setup.cfg
./README.md
./research.ipynb
./.gitignore
```

The **setup.cfg** provides all the relevant metadata that allows the software to be installed from source as a python package, allowing code edits during development at the same time it allows packages to be imported by the package name.

Success Criteria & Evaluation Protocols

This section describes the (a) Primary Metrics which will be observed for the HLM12ERC, (b) The Evaluation protocol used to evaluate it and (c) How Grid Search will be used to evaluate the Final Model.

Primary Metric: Accuracy

One approach (e.g.: Advanced Textual Embeddings) will be considered successful over another if it outperforms the latter in terms of Accuracy[22] over the MELD Test Split.

Hold Out Test Approach

The Evaluation of each component in isolation (Objectives 1 to 6), as well as the final model (for Objective 7) will be carried out with a **Hold Out Test-set** approach[4].

The choice of the Hold Out approach is encouraged by two main factors:

1. The MELD Dataset[1] is provided with three splits: train, dev and test, which already leads to the Hold Out approach;

2. Additionally, the multimodal nature and large number of examples of the MELD dataset make it computationally impossible to utilise approaches such as K-Fold Cross Validation.

Grid Search for Architecture & Hyperparams

Interaction between the components affects the overall results, not visible when each component is investigated individually.

Given that architectural choices go beyond tuning of continuous hyperparameters such as the learning rate, a Grid Search using the Ray library[17] built-into the HuggingFace framework will be used to assess the best settings for the Final model, that allows the hyperparameter space to be defined with categorical options rather than just continuous ranges of values.

Work Plan

The project phases were inspired by the Double Diamond of Interaction Design[21] & the plan can be represented by the **Gantt Chart** from ***Figure 2***.

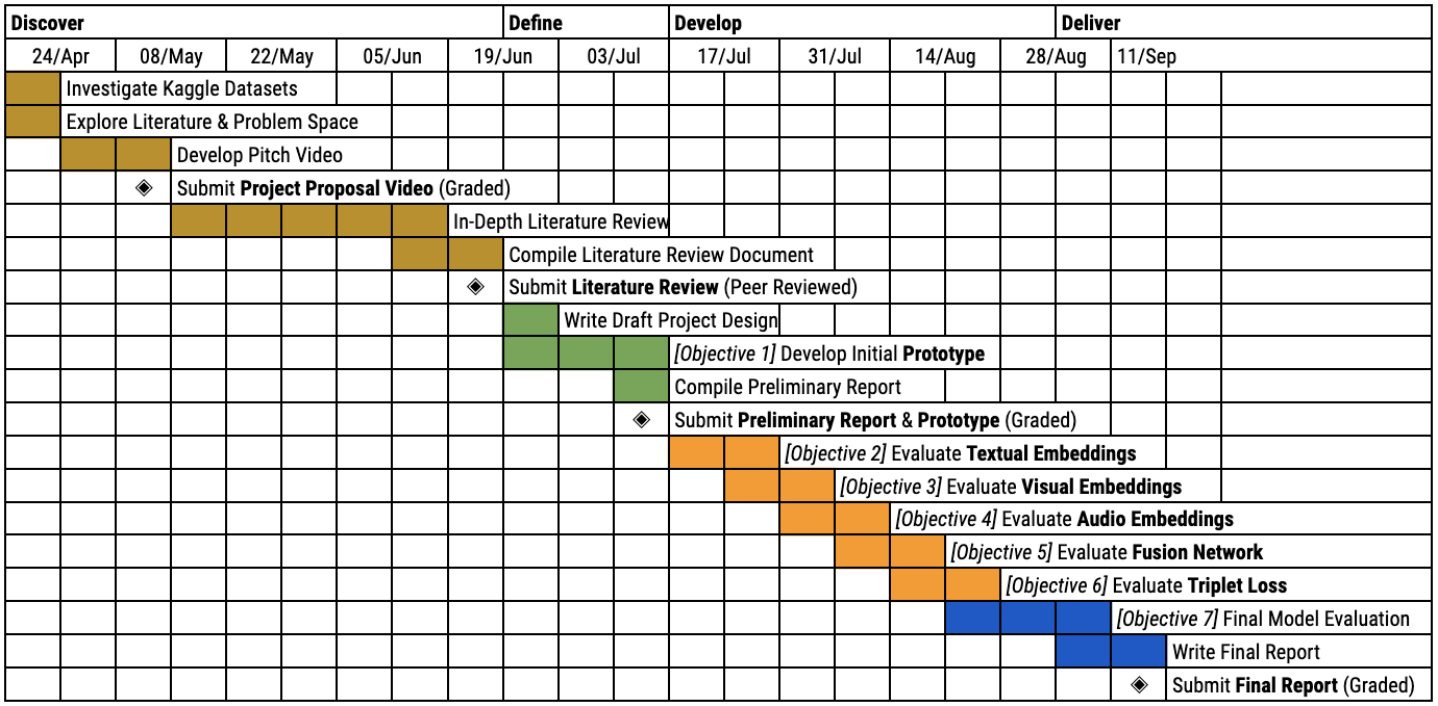

Figure 2: Gantt Chart representation of the Project Plan, including Objectives, Milestones & Phases.

References

[1] Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Gautam Naik, Erik Cambria, and Rada Mihalcea. 2019. MELD: A Multimodal Multi-Party Dataset for Emotion Recognition in Conversations. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 527–536. DOI:https://doi.org/10.18653/v1/P19-1050

[2] Vishal Chudasama, Purbayan Kar, Ashish Gudmalwar, Nirmesh Shah, Pankaj Wasnik, and Naoyuki Onoe. 2022. M2FNet: Multi-modal Fusion Network for Emotion Recognition in Conversation. In 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 4651–4660. DOI:https://doi.org/10.1109/CVPRW56347.2022.00511

[3] Bernhard Kratzwald, Suzana Ilić, Mathias Kraus, Stefan Feuerriegel, and Helmut Prendinger. 2018. Deep learning for affective computing: Text-based emotion recognition in decision support. Decision Support Systems 115, (2018), 24–35. DOI:https://doi.org/https://doi.org/10.1016/j.dss.2018.09.002

[4] Francois Chollet. 2017. Deep Learning with Python (1st ed.), Manning Publications Co., USA.

[5] Xiaohui Song, Longtao Huang, Hui Xue, and Songlin Hu. 2022. Supervised Prototypical Contrastive Learning for Emotion Recognition in Conversation. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 5197–5206. Retrieved from https://aclanthology.org/2022.emnlp-main.347

[6] Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The Long-Document Transformer. Source code: https://github.com/allenai/longformer

[7] MosaicML NLP Team. 2023. Introducing MPT-7B: A New Standard for Open-Source, Commercially Usable LLMs. Retrieved March 28, 2023 from www.mosaicml.com/blog/mpt-7b

[8] Yanjia Zhu, Hongxiang Cai, Shuhan Zhang, Chenhao Wang, and Yichao Xiong. 2021. TinaFace: Strong but Simple Baseline for Face Detection.

[9] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. FaceNet: A unified embedding for face recognition and clustering. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 815–823. DOI:https://doi.org/10.1109/CVPR.2015.7298682

[10] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. Wav2vec 2.0: a framework for self-supervised learning of speech representations. In Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS'20). Curran Associates Inc., Red Hook, NY, USA, Article 1044, 12449–12460.

[11] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In Empirical Methods in Natural Language Processing (EMNLP), 1532–1543. Retrieved from http://www.aclweb.org/anthology/D14-1162

[12] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. Proceedings of the IEEE 86, 11 (1998), 2278–2324. DOI:https://doi.org/10.1109/5.726791

[13] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. ACM Comput. Surv. 55, 9, Article 195 (September 2023), 35 pages. https://doi.org/10.1145/3560815

[14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In Advances in Neural Information Processing Systems, Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[15] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: an imperative style, high-performance deep learning library. Proceedings of the 33rd International Conference on Neural Information Processing Systems. Curran Associates Inc., Red Hook, NY, USA, Article 721, 8026–8037.

[16] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Online, 38–45. Retrieved from https://www.aclweb.org/anthology/2020.emnlp-demos.6

[17] Richard Liaw, Eric Liang, Robert Nishihara, Philipp Moritz, Joseph E Gonzalez, and Ion Stoica. 2018. Tune: A Research Platform for Distributed Model Selection and Training. arXiv preprint arXiv:1807.05118 (2018).

[18] The pandas development team. 2020. pandas-dev/pandas: Pandas. Zenodo. DOI:https://doi.org/10.5281/zenodo.3509134

[19] Brian McFee, Matt McVicar, Daniel Faronbi, Iran Roman, Matan Gover, Stefan Balke, Scott Seyfarth, Ayoub Malek, Colin Raffel, Vincent Lostanlen, Benjamin van Niekirk, Dana Lee, Frank Cwitkowitz, Frank Zalkow, Oriol Nieto, Dan Ellis, Jack Mason, Kyungyun Lee, Bea Steers, Emily Halvachs, Carl Thomé, Fabian Robert-Stöter, Rachel Bittner, Ziyao Wei, Adam Weiss, Eric Battenberg, Keunwoo Choi, Ryuichi Yamamoto, CJ Carr, Alex Metsai, Stefan Sullivan, Pius Friesch, Asmitha Krishnakumar, Shunsuke Hidaka, Steve Kowalik, Fabian Keller, Dan Mazur, Alexandre Chabot-Leclerc, Curtis Hawthorne, Chandrashekhar Ramaprasad, Myungchul Keum, Juanita Gomez, Will Monroe, Viktor Andreevitch Morozov, Kian Eliasi, nullmightybofo, Paul Biberstein, N. Dorukhan Sergin, Romain Hennequin, Rimvydas Naktinis, beantowel, Taewoon Kim, Jon Petter Åsen, Joon Lim, Alex Malins, Darío Hereñú, Stef van der Struijk, Lorenz Nickel, Jackie Wu, Zhen Wang, Tim Gates, Matt Vollrath, Andy Sarroff, Xiao-Ming, Alastair Porter, Seth Kranzler, Voodoohop, Mattia Di Gangi, Helmi Jinoz, Connor Guerrero, Abduttayyeb Mazhar, toddrme2178, Zvi Baratz, Anton Kostin, Xinlu Zhuang, Cash TingHin Lo, Pavel Campr, Eric Semeniuc, Monsij Biswal, Shayenne Moura, Paul Brossier, Hojin Lee, and Waldir Pimenta. 2023. librosa/librosa: 0.10.0.post2. Zenodo. DOI:https://doi.org/10.5281/zenodo.7746972

[20] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. 2020. Array programming with NumPy. Nature 585, 7825 (September 2020), 357–362. DOI:https://doi.org/10.1038/s41586-020-2649-2

[21] Helen Sharp, Jennifer Preece, and Yvonne Rogers. 2019. Interaction Design : Beyond Human-Computer Interaction. John Wiley & Sons, Incorporated, Newark. Retrieved from http://ebookcentral.proquest.com/lib/londonww/detail.action?docID=5746446

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 12, (2011), 2825–2830.