A Cross-Domain Evaluation of Multimodal Argument Relation Identification

Oscar Morris

1 Introduction

2 Background

2.1 Argumentation Theory

Argument and debate has been studied since the time of the ancient Greek philosophers and rhetoricians where argument theorists have sought to formalise discourse and discover some standard of proof for determining the 'correctness' of an argument. Over time, theories of arguments and discussions have evolved, notably when Hamblin [1] refashioned an argumentative discourse as a game, where one party makes moves offering premises that may be acceptable to the another party in the discourse who doubts the conclusion of the argument.

In order to describe various dialogue, argument and illocutionary structures different models (annotation schemes) can be used, some annotation schemes focus on types of the text itself (such as speech act theory [2]) or on the types of relations between components (such as Rhetorical Structure Theory [3]). Inference Anchoring Theory (IAT) [4] is an annotation scheme constructed to benefit from insights across both types, whilst focusing specifically on argumentative discourse. This makes IAT a very useful tool to analyse arguments and their relations.

In IAT, the discourse is first segmented into Argumentative Discourse Units (ADUs). An ADU is any span of the discourse which has both propositional content and discrete argumentative function [5]. An IAT argument graph is typically composed of two main parts: the left-hand side and the right-hand side. The right-hand side is concerned with locutions and transitions between them. A locution is simply the text of the ADU as uttered, without reconstructing ellipses or resolving pronouns. Locutions also include the speaker and may even include a timestamp. Transitions connect locutions capturing a functional relationship between predecessor and successor locutions (i.e. a response or reply).

The left-hand side of an argument graph is more concerned with the content of the ADU, rather than directly reflecting what was uttered. This consists of the propositions made, and the relations between those propositions. To create a proposition from an ADU, the content is reconstructed to be a coherent, lone-standing sentence. This means that any missing or implicit material has to be reconstructed, including anaphoric references (e.g. pronouns).

IAT defines three different types of propositional relation: *inference*, *conflict* and *rephrase*. An inference relation (also termed RA) holds between two propositions when one (the premise) is used to provide a reason to accept the other (the conclusion). This may include annotation of the kind of support e.g. Modus Ponens or Argument from Expert Opinion. These subtypes of relation are often called *argument schemes* [6], [7]. There are also several different inference structures:

- **Serial arguments** occur when one proposition supports another, which in turn supports a third.

- **Convergent arguments** occur when multiple premises act independently to support the same conclusion.
- **Linked arguments** occur when multiple premises work together to support a conclusion.
- **Divergent arguments** occur when a single premise is used to support multiple conclusions.

A conflict relation (also termed CA) holds between two propositions when one is used to provide an incompatible alternative to another and can also be of a given kind (e.g. Conflict from Bias, Conflict from Propositional Negation). The following conflict structures are identified by IAT:

- **Rebutting conflict** occurs if one proposition is directly targeting another by indicating that the latter is not acceptable.
- **Undermining conflict** occurs if a conflict is targeting the premise of an argument, then it is undermining its conclusion.
- **Undercutting conflict** occurs if the conflict is targeting the inference relation between two propositions.

A rephrase relation (also termed MA) holds when one proposition rephrases, restates or reformulates another but with dfferent propositional content (i.e. one proposition cannot simply repeat the other). There are many different kinds of rephrase, such as Specialisation, Generalisation, Instantiation etc. Generally, question answering will often involve a rephrase because the propositional content of the question is typically instantiated, resolved or refined by its answer. In contrast to inference, conflict and rephrase structures only have a single incoming an one outgoing edge.

The left and right-hand sides are connected by *illocutionary connections*. These illocutionary connections are based on illocutionary force as introduced by speech act theory [2]. The speech act $F(p)$ is the act which relates the locution and and the propositional content p through the illocutionary force F e.g. asserting p, requesting p, promising p etc. There are many diferent types of illocutionary connection, including: assertions, questions, challenges, concessions and (dis-)affirmations [8].

There have been several ways to store argumentative data created, for example Argument Markup Language (AML) [9], an XML-based language used to describe arguments in the Araucaria software. More recently, the Argument Interchange Format (AIF) [10] has been created to standardise the storage of IAT graphs.

AIF treats all relevant parts of the argument as nodes within a graph. These nodes can be put into two categories: *information nodes* (I-nodes) and *scheme nodes* (S-nodes). I-nodes represent the claims made in the discourse whereas S-nodes indicate the application of an argument scheme. Initially I-nodes only included the propositions made [10], but when Reed *et al.* [11] extended AIF to cater to dialogues, they added L-nodes as a subclass of I-nodes to represent locutions. For the purposes of this research, I-nodes and L-nodes are considered separate classes where I-nodes contain propositions and L-nodes contain locutions.

Since AIF data can be easily shared, it became the basis for a Worldwide Argument Web (WWAW) [12]. Since then, many corpora have been annotated using IAT and published on the AIFdb[1] [13] providing a very useful resource for argumentation research of many kinds.

2.2 Machine Learning

In recent years there have been several major advances in the field of natural language processing (NLP), most notably the introduction of the transformer architecture [14]. The transformer architecture, based on self-attention,

[1] https://www.aifdb.org/

allows the model to determine much longer range dependencies than previous approaches.

Even before Vaswani *et al.* introduced the transformer architecture supervised and semi-supervised pre-training approaches were already being explored, and proven to be a very useful tool for improving the performance of language models [15], [16]. When the transformer was introduced these pre-training techinques were adapted for use in transformers creating models which are able to be fine-tuned with relatively minimal effort and compute to allow high performance on a wide variety of tasks [17], [18]. The pre-training approaches introduced by BERT and RoBERTa use a combination of masked language modelling (where the model is trained to predict the token hidden under a `[mask]` token) and next sentence prediction. The models are then trained using this approach on a large amount of data (the data used to pre-train RoBERTa totals over 160GB of uncompressed text).

A similar progression can be seen in the development of audio models. Pre-training was notably introduced into speech recognition with wav2vec [19], where the model is trained to predict future samples from a given signal. The wav2vec model has two main stages, first raw audio samples are fed into a convolutional network which performs a similar role to the tokenisation seen in text-based language models by using a sliding window approach to downsample the audio data. These encodings are then fed into a second convolutional network to create a final encoding for the sequence.

Transformer models were introduced into the architecture of audio models with wav2vec2 [20] and HuBERT [21], where the second convolutional model is replaced with a transformer in order to better learn dependencies across the entire sequence. These models are then pre-trained on significant amounts of audio data (960 hours in the case of wav2vec2) in order to then be fine-tuned on a downstream task.

Combining modalities (such as text and audio) has also proven to be a useful tool across several tasks, including medical imaging [22] and natural language processing [23], [24], including argument mining [25], [26]. Generally fusion techniques can be split into two categories: early and late. Early fusion techniques combine representations of each modality before being used as input to an encoder, with the primary benefit that only a single encoder is used. Late fusion techniques use a separate encoder for each modality, and the encodings are then fused to provide a crossmodal representation of the input.

In early fusion the input representations are transformed into a common information space, often using vectorisation techniques dependent on the modality.

2.3 Argument Mining

3 Datasets

All datasets used in this project are available as corpora on AIFdb[2]. Using consistently annotated Argument Interchange Format (AIF) data allows many different datasets to be used and tested. The AIF Format [10] allows the annotation of argument data across all AM tasks, providing a platform for many different kinds of research.

Throughout the project two primary corpora have been considered: QT30 [27], a corpus consisting of 30 AIF annotated Question Time episodes, and a corpus of 9 AIF annotated Moral Maze episodes available on AIFdb.

3.1 Preprocessing

3.1.1 Argument Data

In order to use AIF data efficiently for ARI, it is useful to perform some preprocessing. This process produces a graph, where each node

[2]https://corpora.aifdb.org/

contains a locution, its related proposition, and the proposition's AIF identifier. This identifier corresponds to the audio data, allowing it to be easily loaded when required. Each edge in this graph corresponds to a relation between the propositions, one of RA (inference), MA (rephrase) or CA (conflict).

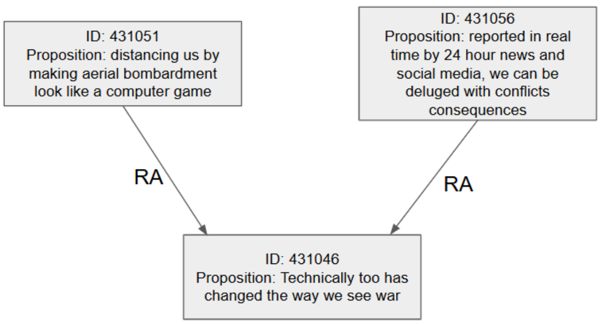

Figure 1: Example sub-graph.

Figure 1 shows an example sub-graph from the larger argument graph. Each node is truncated for brevity and only shows the node's ID, and the proposition. This sub-graph is taken from the Moral Maze episode on the 75th Anniversary of D-Day.

An example of the JSON structure used to store the argument data is shown in Listing 3.1.1.

```
{
    "id": 433407,
    "locution": "Matthew Taylor : she answered questions about norms and structures by talking about beliefs and campaigns and I think beliefs are different to norms and I think campaigns are different to social structures",
    "proposition": "Nancy Sherman answered questions about norms and structures by talking about beliefs and campaigns and beliefs are different to norms and campaigns are different to social structures",
    "relations": [
        {
            "type": "CA",
            "to_node_id": 433393
        },
        {
            "type": "RA",
            "to_node_id": 433416
        }
    ],
}
```

An array of these JSON objects can then be used to create the node pairs required for the training and evaluation of the model.

3.1.2 Audio Data

The audio data first had to be downsampled from 44.1kHz to the 16kHz which is best accepted by the Wav2Vec2 transformer [20] among many others. This can easily be achieved using FFmpeg[3]. In the case of QT30, first audio had to be extracted from the video, and collapsed into a mono track before it could be downsampled, this was also easily achieved with FFmpeg.

Next, start and end times for each locution in the argument graph need to be found, to allow the audio to be split per-locution (and therefore per node). This can be achieved using PyTorch's forced alignment api[4].

Initially this was achieved by aligning each word in the transcript of the episode, producing start and end times for each word. A search can then be performed through this data to find the required locution. While this technique initially produced promising results, it was not robust enough to allow for errors in the transcripts or the crosstalk common in debates.

To solve this problem, the PyTorch forced alignment api is able to take wildcard tokens as input, therefore, each locution can be searched

[3]https://ffmpeg.org/
[4]https://pytorch.org/audio/

for individually. To achieve this, the partial transcript used as input to the forced aligner took the following form: `* {locution} *`.

Using this system allows the forced aligner to work well through crosstalk (since each locution's alignments are searched for independently of all others), and qualitatively seems to be more resilient to errors. Error resilience is helped since errors are less common in the location texts as opposed to the transcripts. Using this system also allowed for confidence scores to be collected for analysis. In this section general analysis across all corpora is performed, with corpus specific analysis in the relevant section.

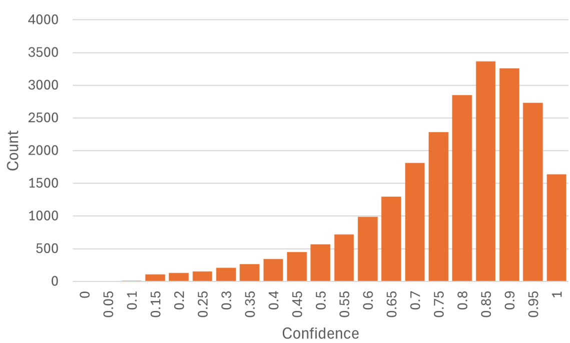

Figure 2: Confidence distribution across all corpora.

Figure 2 shows the distribution of confidence scores across both the QT30 corpus and all Moral Maze corpora. This distribution shows that the system can relatively confidently align the majority of locutions, with only approx. 8% of locutions with a confidence score less than 0.50.

In order to further analyse the performance of this system, locutions were selected at random and qualitatively analysed. Throughout this process, all locutions appeared correct, however, it was very challenging to accurately determine the accuracy of the system on locutions with confidence scores < 0.2. This shows that this method of aligning locutions with their corresponding audio is accurate.

3.1.3 Pair Creation

Finally, a set of node pairs and their relations can be generated in order to train a neural network. For related nodes this can be done trivially in that for each relation, the corresponding pair of nodes can be added to the set. When sampling unrelated nodes, however, things are more complex.

For this project, Short Context Sampling (SCS) is used as presented in [28]. Given the episodic structure of both QT30 and the Moral Maze corpora, a short context can be defined as the episode. This also allows the model to learn in a more realistic environment. Since the vast majority of node pairs have no relation, a number equal to that of Inference relations are generated.

3.2 QT30

The QT30 argument corpus [27] contains transcripts and argument annotations for 30 episodes of the BBC's Question Time, a series of televised topical debates across the United Kingdom. All episodes aired in 2020 and 2021. The corpus is split into 30 subcorpora, each spanning a single episode. This allows analysis of each episode individually, or combined as a single corpus.

Table 1: Disribution of propositional relations in QT30.

Relation Type	Count	Proportion (%)
Inference	5,761	51.4%
Conflict	947	8.5%
Rephrase	4,496	40.1%
Total	11,204	100%

Table 1 shows the distribution of each type of relation across QT30. Inference and Rephrase relations make up a total of 91.5% of the dataset, with Conflict relations being significantly less common, only making up 8.5% of the dataset. It is obvious that this is an unbalanced dataset, which will have to be considered during training.

Table 2: Mean confidence scores (μ) and standard deviation of confidence scores (σ) across each QT30 subcorpus.

Corpus Name	μ	σ
28May2020	0.76	0.16
4June2020	0.72	0.17
18June2020	0.76	0.16
30July2020	0.75	0.16
2September2020	0.78	0.15
22October2020	0.76	0.16
5November2020	0.77	0.17
19November2020	0.74	0.19
10December2020	0.77	0.16
14January2021	0.74	0.17
28January2021	0.70	0.17
18February2021	0.75	0.16
4March2021	0.76	0.16
18March2021	0.75	0.17
15April2021	0.75	0.15
29April2021	0.70	0.19
20May2021	0.76	0.17
27May2021	0.79	0.15
10June2021	0.75	0.17
24June2021	0.74	0.17
8July2021	0.72	0.17
22July2021	**0.44**	**0.28**
5August2021	0.76	0.17
19August2021	0.77	0.17
2September2021	0.78	0.16
16September2021	0.77	0.16
30September2021	**0.24**	**0.08**
14October2021	0.75	0.19
28October2021	0.75	0.17
11November2021	0.78	0.16
QT30	0.75	0.17

Table 2 shows the mean and standard deviation of the confidence scores across each of the QT30 subcorpora. The lowest two mean scores are shown in bold. Manually analysing samples in these episodes indicates a high error rate in the alignment of locutions. Because of this high error rate, it was decided to exclude these episodes from the corpus used for training. The excluded episodes are: 22July2021 and 30September2021. The rest of the episodes

Table 3: Disribution of propositional relations in QT30-MM.

Relation Type	Count	Proportion (%)
Inference	5,740	51%
Conflict	937	8.4%
Rephrase	4,479	40.6%
Total	11,156	100%

from QT30 will form its multimodal subcorpus (QT30-MM).

Similarly to the complete QT30 corpus, Inference and Rephrase make up the vast majority of the QT30-MM dataset, with the proportion of Conflict relations decreasing to 8.4%.

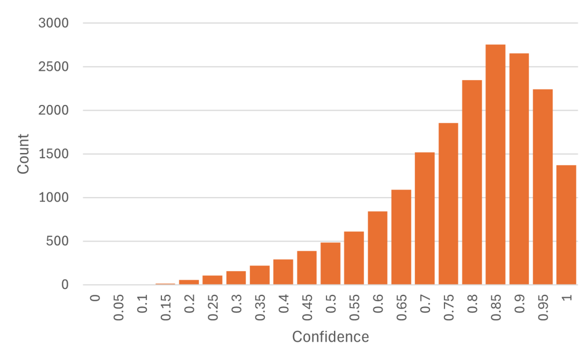

Figure 3: Confidence distribution across QT30-MM.

As can be seen in Figure 3 the distribution of confidence scores closely matches that shown in Figure 2. This still indicates that the audio alignments are calculated with high accuracy.

Table 4: Distribution of propositional relations after sampling non-related nodes.

Relation Type	Count	Proportion (%)
None	5,470	34%
Inference	5,470	34%
Conflict	937	6%
Rephrase	4,479	27%
Total	16,896	100%

Table 5: Distribution of propositional relations across the Moral Maze corpus.

Relation Type	None	Inference	Conflict	Rephrase	Total
B	132 (45%)	132 (45%)	24 (8%)	3 (1%)	291
E	151 (41%)	151 (41%)	39 (11%)	25 (7%)	366
M	255 (43%)	255 (43%)	29 (5%)	58 (10%)	597
P	236 (42%)	236 (42%)	40 (7%)	45 (8%)	557
S	181 (42%)	181 (42%)	63 (14%)	10 (2%)	435
G	301 (41%)	301 (41%)	46 (6%)	93 (13%)	741
D	72 (40%)	72 (40%)	7 (4%)	28 (16%)	179
H	207 (43%)	207 (43%)	23 (5%)	43 (9%)	480
Total	1,535 (42%)	1,535 (42%)	271 (7%)	305 (8%)	3,646

3.3 Moral Maze

Similar to Question Time, the BBC's Moral Maze is a series of radio broadcast debates, with each episode focusing on a certain topic. Seven different Moral Maze episodes have been AIF annotated and made available on AIFdb. It is therefore these seven episodes, released from 2012 to 2019, which this project considers. Each episode focuses on a very different domain which allows for a robust, cross-domain analysis of any models trained on another corpus (e.g. QT30). The Moral Maze corpus contains data from nine different episodes: Banking (B), Empire (E), Money (M), Problem (P), Syria (S), Green Belt (G), D-Day (D) and Hypocrisy (H). Each episode consists of a debate focusing on a different topic, and hence has a different distribution of classes.

Table 5 shows the distribution of propositional relations across the Moral Maze corpus, after non-related pairs have been sampled. Comparing the corpus to QT30, a significantly lower proportion of the corpus is made up of Rephrase relations. It is possible that the differing formats of the debates has an impact here.

In Table 6 the mean and standard deviation of audio alignemnt confidence scores are compared across subcorpora. Generally the results match what is expected and are similar to those in QT30, the system does achieve unusually low scores when considering the D-Day subcorpus, the reason for this is unclear, however, manually analysing both random and low-confidence samples indicates they are generally correct and so the subcorpus can be used for the cross-domain evaluation.

Table 6: Mean confidence scores (μ) and Standard Deviation of confidence scores (σ) across Moral Maze subcorpora.

Subcorpus	μ	σ
B	0.79	0.14
E	0.76	0.15
M	0.78	0.14
P	0.80	0.15
S	0.74	0.16
G	0.80	0.14
D	**0.50**	**0.34**
H	0.74	0.16

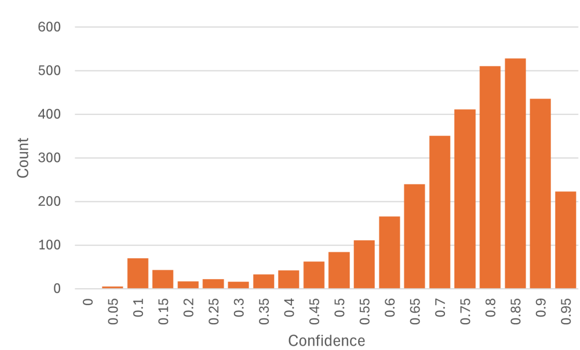

Figure 4: Confidence distribution across all Moral Maze subcorpora.

Figure 4 shows the distribution of audio alignment confidence scores across all Moral Maze episodes. This follows the expected pattern as shown in Figures 2 and 3. The secondary peak around a confidence of 0.10 is caused by the D-Day subcorpus.

4 Models

There are many different approaches when considering multimodal models. Generally they can be split into two categories: early fusion and late fusion. Early fusion models initially combine the features from each modality together, before being used as input to a single transformer model. Late fusion models use multiple transformers, one specialised in each modality. After being fed through each transformer, the hidden vectors are combined before being fed into the model's head. Initially, only late fusion models have been considered in this research.

This late fusion model uses RoBERTa-base [18] as its text transformer. To process the audio data, the model uses the Wav2Vec2-base audio transformer [20], having been pre-trained on 960 hours of dialogue[5]. The base models are used currently in order to increase the speed at which experiments can be conducted, however, they are used with a view to transitioning to their large variants towards the end of the project.

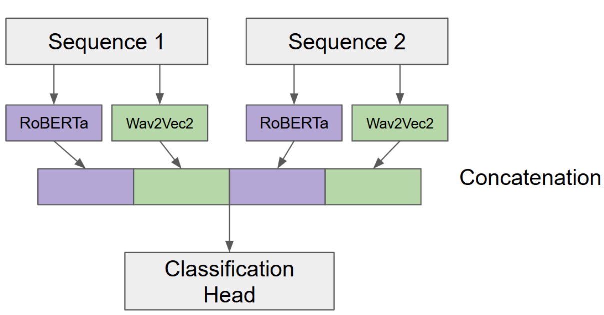

Figure 5: Concatenation model with late fusion.

[5]https://huggingface.co/facebook/wav2vec2-base-960h

Figure 5 povides a visualisation of the primary model used at this point in the project. Similar models have been shown to achieve good multimodal performance while also being relatively simple to implement [26], [29]. It is for these reasons that this model has been implemented using the Huggingface[6] and PyTorch[7] python modules.

Some preliminary experiments have also been conducted using text-only models in an attempt to replicate results produced in [30]. For this the RoBERTa-base model is used and each sentence is concatenated before tokenisation, delimited by RoBERTa's special purpose token `</s>`.

5 Results

5.1 In-Dataset

5.2 Cross-Dataset

6 Limitations

7 Conclusions

8 Future Work

References

[1] C. L. Hamblin, *Fallacies*, First Edition. London: Methuen young books, 1970.

[2] J. R. Searle, *Speech Acts: An Essay in the Philosophy of Language*. Cambridge: Cambridge University Press, 1969. doi: 10.1017/CBO9781139173438.

[6]https://huggingface.co/

[7]https://pytorch.org/

[3] W. C. Mann and S. A. Thompson, “Rhetorical Structure Theory: Toward a functional theory of text organization,” *Text - Interdisciplinary Journal for the Study of Discourse*, vol. 8, no. 3, pp. 243–281, Jan. 1988, doi: 10.1515/text.1.1988.8.3.243.

[4] C. Reed and K. Budzynska, “How dialogues create arguments,” 2011.

[5] C. Reed, “A Quick Start Guide to Inference Anchoring Theory (IAT).” Sep. 2017.

[6] D. Walton, C. Reed, and F. Macagno, *Argumentation Schemes*. Cambridge: Cambridge University Press, 2008. doi: 10.1017/CBO9780511802034.

[7] D. Walton, “Argumentation Theory: A Very Short Introduction,” in *Argumentation in Artificial Intelligence*, G. Simari and I. Rahwan, Eds. Boston, MA: Springer US, 2009, pp. 1–22. doi: 10.1007/978-0-387-98197-0_1.

[8] K. Budzynska, M. Janier, C. Reed, P. Saint-Dizier, M. Stede, and O. Yakorska, “A Model for Processing Illocutionary Structures and Argumentation in Debates,” in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC‘14)*, May 2014, pp. 917–924.

[9] C. Reed and G. Rowe, “Araucaria: Software for argument analysis, diagramming and representation,” *International Journal on Artificial Intelligence Tools*, vol. 13, no. 4, pp. 961–979, Dec. 2004, doi: 10.1142/S0218213004001922.

[10] C. Chesñevar *et al.*, “Towards an argument interchange format,” *The Knowledge Engineering Review*, vol. 21, no. 4, pp. 293–316, Dec. 2006, doi: 10.1017/S0269888906001044.

[11] C. Reed, J. Devereux, S. Wells, and G. Rowe, “AIF+: Dialogue in the Argument Interchange Format.”

[12] I. Rahwan, F. Zablith, and C. Reed, “Laying the foundations for a World Wide Argument Web,” *Artificial Intelligence*, vol. 171, no. 10–15, pp. 897–921, Jul. 2007, doi: 10.1016/j.artint.2007.04.015.

[13] J. Lawrence, F. Bex, C. Reed, and M. Snaith, “AIFdb: Infrastructure for the Argument Web,” in *Computational Models of Argument*, IOS Press, 2012, pp. 515–516. doi: 10.3233/978-1-61499-111-3-515.

[14] A. Vaswani *et al.*, “Attention Is All You Need,” p. 11, 2017.

[15] M. E. Peters *et al.*, “Deep contextualized word representations.” arXiv, Mar. 2018. doi: 10.48550/arXiv.1802.05365.

[16] A. M. Dai and Q. V. Le, “Semi-supervised Sequence Learning.” arXiv, Nov. 2015. doi: 10.48550/arXiv.1511.01432.

[17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.” arXiv, May 2019. doi: 10.48550/arXiv.1810.04805.

[18] Y. Liu *et al.*, “RoBERTa: A Robustly Optimized BERT Pretraining Approach.” arXiv, Jul. 2019. doi: 10.48550/arXiv.1907.11692.

[19] S. Schneider, A. Baevski, R. Collobert, and M. Auli, “Wav2vec: Unsupervised Pre-training for Speech Recognition.” arXiv, Sep. 2019. Accessed: Oct. 17, 2024. [Online]. Available: https://arxiv.org/abs/1904.05862

[20] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, “Wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations.” arXiv, Oct. 2020. Accessed: Oct. 16, 2024. [Online]. Available: https://arxiv.org/abs/2006.11477

[21] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units." arXiv, Jun. 2021. doi: 10.48550/arXiv.2106.07447.

[22] J. Delbrouck *et al.*, "ViLMedic: A framework for research at the intersection of vision and language in medical AI," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, May 2022, pp. 23–34. doi: 10.18653/v1/2022.acl-demo.3.

[23] E. Toto, M. Tlachac, and E. A. Rundensteiner, "AudiBERT: A Deep Transfer Learning Multimodal Classification Framework for Depression Screening," in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, Oct. 2021, pp. 4145–4154. doi: 10.1145/3459637.3481895.

[24] Y.-H. H. Tsai, S. Bai, P. P. Liang, J. Z. Kolter, L.-P. Morency, and R. Salakhutdinov, "Multimodal Transformer for Unaligned Multimodal Language Sequences," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Jul. 2019, pp. 6558–6569. doi: 10.18653/v1/P19-1656.

[25] R. Mestre, R. Milicin, S. E. Middleton, M. Ryan, J. Zhu, and T. J. Norman, "M-Arg: Multimodal Argument Mining Dataset for Political Debates with Audio and Transcripts," in *Proceedings of the 8th Workshop on Argument Mining*, Nov. 2021, pp. 78–88. doi: 10.18653/v1/2021.argmining-1.8.

[26] E. Mancini, F. Ruggeri, S. Colamonaco, A. Zecca, S. Marro, and P. Torroni, "MAMKit: A Comprehensive Multimodal Argument Mining Toolkit," in *Proceedings of the 11th Workshop on Argument Mining (ArgMining 2024)*, Aug. 2024, pp. 69–82. doi: 10.18653/v1/2024.argmining-1.7.

[27] A. Hautli-Janisz, Z. Kikteva, W. Siskou, K. Gorska, R. Becker, and C. Reed, "QT30: A Corpus of Argument and Conflict in Broadcast Debate," in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, Jun. 2022, pp. 3291–3300.

[28] R. Ruiz-Dolz, D. Gemechu, Z. Kikteva, and C. Reed, "Looking at the Unseen: Effective Sampling of Non-Related Propositions for Argument Mining," in *Proceedings of the 31st International Conference on Computational Linguistics*, Jan. 2025, pp. 2131–2143.

[29] E. Mancini, F. Ruggeri, A. Galassi, and P. Torroni, "Multimodal Argument Mining: A Case Study in Political Debates," in *Proceedings of the 9th Workshop on Argument Mining*, Oct. 2022, pp. 158–170.

[30] D. Gemechu, R. Ruiz-Dolz, and C. Reed, "ARIES: A General Benchmark for Argument Relation Identification," in *Proceedings of the 11th Workshop on Argument Mining (ArgMining 2024)*, Aug. 2024, pp. 1–14. doi: 10.18653/v1/2024.argmining-1.1.