A Heuristic Method for Automatic Gaze Detection in Constrained Multi-Modal Dialogue Corpora

Lorcan McLaren
Computational Linguistics Group
Trinity College Dublin
The University of Dublin
Ireland
Email: lmclaren@tcd.ie

Maria Koutsombogera
Computational Linguistics Group
Trinity College Dublin
The University of Dublin
Ireland
Email: koutsomm@scss.tcd.ie

Carl Vogel
Computational Linguistics Group
Trinity College Dublin
The University of Dublin
Ireland
Email: vogel@tcd.ie

***Abstract*—We describe a heuristic-based approach to determining gaze allocation automatically in a multi-modal task oriented dialogue corpus. We present the development of the system and the evaluation of its performance and discuss the findings, including the shortcomings and the perspectives of the implemented approach.**

I. Introduction

In this work we describe heuristics that enable automated gaze annotation in order to link gaze as recorded in a multi-modal corpus with interactional qualities. The multi-modal corpus adopted is the MULTISIMO corpus of task based dialogues, each involving two participants collaborating with each other in the company of a facilitator to estimate popular opinions. The motivation behind this approach was to create a system that works out-of-the-box for triadic groups conforming to the MULTISIMO experimental setup, and that requires no training data that might otherwise have to be manually produced. Advantages of a rule-based system include transparency behind classifications and the ability to fine-tune the heuristics to optimise accuracy across participants – a process would require retraining a model in a machine learning-based approach. The resulting annotations are available to support future research with the MULTISIMO corpus.

This work contributes to cognitive infocommunications research [1]–[3], particularly the thread which attends to linguistic and behavioural interaction [4], where there is interest in studying gesture and facial expression alongside linguistic behaviours as recorded in multi-modal data sets, without special purpose eye-trackers or the like [5]–[9]. The field may benefit from using a similar approach to automatic gaze annotation in support of such analyses. The remainder of this paper is organised as follows: §II addresses more of the relevant literature; §III describes the corpus and development of the annotation system; §IV evaluates the performance of the approach; analysis of the data generated; §V discusses these findings; §VI offers a conclusion and some perspectives on possible future work.

II. Related Work

Kendon theorises that gaze serves three primary functions in social interaction: monitoring, expression and regulation [10]. Monitoring describes gaze used as a source of nonverbal information from interlocutors, whereas expressive gaze conveys information to interlocutors – for example, it is suggested that mutual gaze duration is used to generate and manage intimacy [11]. Gaze also serves to regulate processes necessary for successful interaction, including indicating addressee and managing turn-taking. For example, in turn-taking, the speaker will link their utterance to what was previously said. They will then gaze away while they continue to hold the floor, before gazing back towards the hearer as they [the speaker] provide a new piece of information. This gaze behaviour and content together indicate the end of the turn [12].

Much of the past and current work on gaze tracking makes use of eye-tracking glasses, motion-capture cameras or stereo camera setups [13]. Such systems are necessary where highly accurate tracking is required, however, the proposed system relies only on the availability of a single front-facing camera per participant and can be used to provide broad gaze annotation for corpora where facility for automatic annotation had not previously been considered.

Other work has explored using machine learning approaches to automatic gaze annotation. Fukuhara and Nakano [14] use decision trees to make inferences about gaze direction based on head motion data in their Wizard of Oz experiment. The VACE corpus uses 2D data to determine head posture, unless the subject faces the camera directly in which case head posture is inferred from a dual-camera setup using annealed particle filtering [15]. More recently, deep learning approaches using CNNs have been proposed [16]. There remains a need to be able to analyze data for which there is not a sufficient supply to anticipate that deep learning will supply adequate accuracy. Especially where knowledge about the recording setup of a multi-modal dialogue corpus is available, it is sensible to exploit that knowledge directly.

III. Materials & Methods

This section describes the methods involved in this research, including an overview of the corpus and the software libraries used, the processes involved in the video analysis and the subsequent creation of annotations.

A. Resources

The research described in this paper makes use of the MULTISIMO corpus [17] as well as the ELAN editor [18] and two Python libraries used in the video analysis component of the automatic generation of gaze annotation.

MULTISIMO is a multimodal, multiparty corpus that enables observations related to verbal and non-verbal behavior during three-party task-based dialogues. The corpus is made up of sessions involving three participants, with two participants acting as players and the other acting as facilitator of the session. The participants play a game intended to elicit cooperation among the players, similar to the popular American game show *Family Feud*. The facilitator asks a series of questions and players must come up with the 3 responses for each, ranked from most likely to least likely, that they believe will be the most common answers in a survey of 100 individuals when asked the same question. The facilitator provides feedback and guidance until the correct responses and order are achieved.

The corpus contains audio and video recordings from 18 different sessions involving 36 randomly allocated pairs of players, and 3 individuals who act as facilitators for each session. High quality video files with a resolution of 960x540 were used for the three participants. All cameras involved in the setup shoot at 30 frames per second. Audio is captured by an omnidirectional microphone and the head mics of each participant. Figure 1 shows the positions of facilitator, players and cameras for each session. A variety of different annotations are available for the corpus sessions including speech transcription, gesture, laughter and facilitator feedback annotation. Annotations performed in ELAN have 4 important components: the start time, the end time and the duration of the period of the annotation, as well as the actual annotation value itself. Automated gaze annotation provided here may also be viewed and analysed in ELAN.

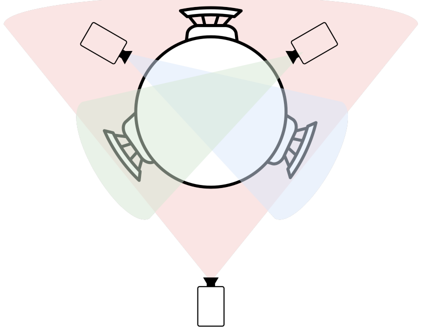

Fig. 1. Plan of the room setup in MULTISIMO sessions

1) Existing Gaze Annotation: Manual gaze annotation already existed for two corpus sessions as a result of previous work [19]. The goal of the video analysis and annotation steps of this research is to replicate the format of these existing annotations as faithfully as possible, while also using them as a reference for validating the automatically generated annotations – though existing annotations will not necessarily be used as a gold-standard of accuracy for reasons that will be discussed later. The existing gaze annotation uses a controlled vocabulary, i.e. a predefined list of possible annotation values for each of the players: GAZE_PLAYER, GAZE-FACILITATOR, GAZE_AWAY. An annotation tier also exists for the facilitator, including the GAZE_AWAY annotation value and 2 values relating to the subject of the facilitator's gaze: GAZE_PLAYER-LEFT, GAZE_PLAYER-RIGHT. Throughout this paper, left and right will correspond to the perspective of a viewer who faces the facilitator/player in question.

B. Video Analysis

OpenCV is an open source image processing library built-in C++ that is intended to provide access to powerful computer vision techniques that work in near real time [20]. Images in OpenCV are represented as two-dimensional matrices of pixels. In RGB colour space, this element will be a tuple where each value represents a shade of red, green and blue respectively. Many computer vision techniques only function in grayscale colour space, where each pixel is represented by a single integer value, thereby reducing dimensionality that is prohibitively expensive in terms of memory and processing time for complex operations. Video in OpenCV is analysed as a sequence of images, and therefore all the operations described in the following sections must be applied to each frame individually. This leads to results that are sometimes imperfect as each image is analysed without taking the preceding or succeeding frames in account. However, as discussed in §III-C1, with some filtering, we can reduce the overall error rate and improve the quality of the resulting annotations.

Dlib is a C++ software library that provides implementations of machine learning tools and pretrained models for a wide variety of applications including image processing. The frontal face detector followed by the shape predictor are used to project a number of facial landmarks onto the located face that will be used for all of the steps described below. These 68 facial landmarks allow for the tracking and isolation of certain points and regions of the face.

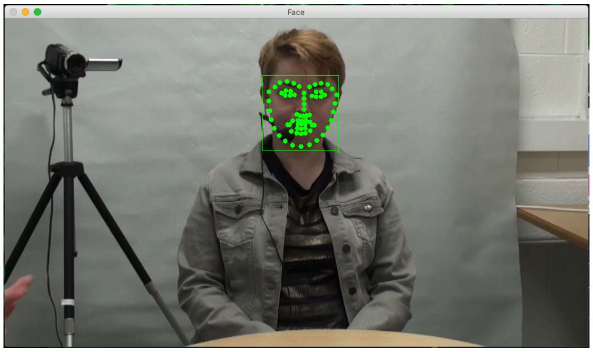

Fig. 2. Dlib's facial landmarks in action

The most significant landmarks for the subsequent stages of analysis and their relevant index were: Chin tip – 8; Nose tip – 33; Left eye region – 36-41; Right eye region – 42-47; Mouth corners –48 & 54. Visual information is among the most unconstrained and saturated forms of data to deal with and performance of computer vision solutions are impacted by lighting conditions and angles of seats among other factors.

1) Blink Detection: Each eye region was isolated by cropping each frame using the location of the landmarks described above. The Euclidean distance between the two landmarks indicating the inner and outer corner of the eye was calculated.

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{1}$$

No landmarks indicate the centre of the upper and lower eyelid, but this was approximated by calculating the midpoint between the two landmarks provided for each lid. Once again, the Euclidean distance between the two midpoints was found. Dividing the first distance by the second produces a ratio that we refer to as the ‘blinking ratio’. Using a blinking ratio rather than a simple pixel count between upper and lower lids ensures that blinking detection is scale-independent and will function correctly regardless of the resolution of the source video or the proximity of the participant to the camera. This ratio was calculated independently for each eye and then averaged. A ‘blinking’ classification was assigned to a frame if the ratio value exceeded a specified threshold. Blink detection was a necessary step both for preventing an error where the eyes could not be found if closed in a given frame – which caused the gaze direction analysis step to fail – as well as for the reason that this data may prove to be useful in the context of this and/or future research.

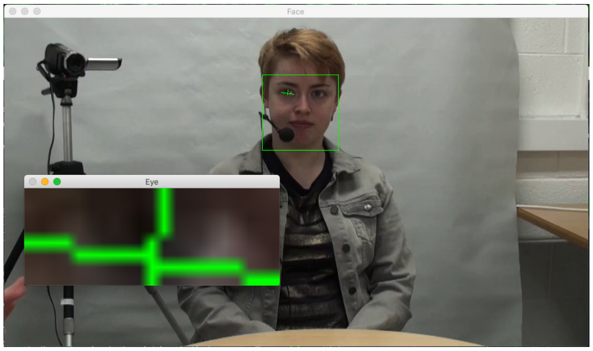

Fig. 3. Calculating the blinking ratio

2) Gaze Direction Analysis: Once again, each eye region was isolated using the facial landmarks in the same manner as the previous step. Initially, an attempt was made to mask the region surrounding the eye itself in order to exclude parts of the lids that had been included when the eye was isolated. However, based on the observation of test runs via webcam and clips from MULTISIMO sessions, early performance needed to be improved, and removing this mask was one change that led to a marked improvement.

The image was then thresholded to separate the two regions of interest in the eye – the iris and the sclera (‘white’ of the eye) – by creating a binary image. Binary thresholding is an operation where each pixel in an image is mapped to the maximum or minimum possible value (white and black respectively, in grayscale colour space) based on its initial value in relation to a specific threshold. For example, if a threshold of 127 were chosen, all pixels with a value above the threshold of the grayscale spectrum would be mapped to white while all values below the threshold would be mapped to black. Otsu’s method dynamically determines an optimum threshold based on the image histogram. It performs well for ‘bimodal’ images, i.e. images where a high contrast exists between the regions we wish to separate. In OpenCV, the threshold value chosen is the one that maximises the between class variance σ_B^2 for the ‘foreground’ and ‘background’ regions [21] i.e. the iris and sclera in our case.

$$\sigma_B^2(T) = w_f(T)(\mu_f(T) - \mu)^2 + w_b(T)(\mu_b(T) - \mu)^2 \quad (2)$$

The above equation is used to calculate this between class variance for the f (foreground) and b (background) classes, where T is the threshold value and μ and σ^2 are the mean value and variance of the image respectively.

A number of operations were used to refine the newly formed binary regions. Erosion reduces the number of object pixels by removing pixels from the perimeter of the foreground, ensuring a precise edge around the region of interest. Dilation expands the number of object pixels, filling any small gaps that may appear within the region of interest. Together, these two operations improved the quality of the binary image by ensuring the foreground and background were cleanly separated into uninterrupted regions with precise boundaries. The eye region was then split into left and right halves in a similar manner to that used for blinking detection. The midpoints between the two landmarks on each lid were calculated separately and the line between these two midpoints was used to divide the eye region.

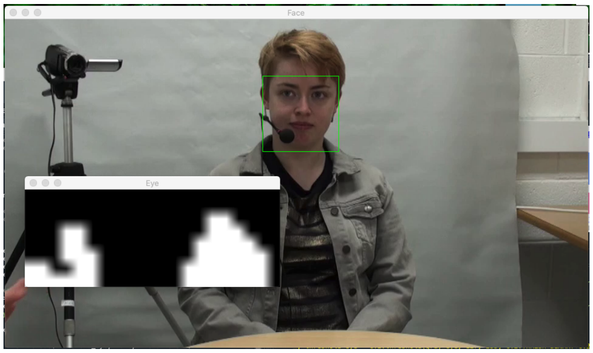

Fig. 4. Thresholding the eye region

It was then necessary to count non-zero elements in each half. Since white has a value of 255 and black has a value of 0 in grayscale colour space, this is equivalent to counting the white pixels. Finally, the ratio of the returned value for the left side of the eye with respect to that of the right side was found.

Gaze direction was classified into three ordinal directions – left, centre and right – on the basis of this value through use of an upper and lower threshold. If the ratio value was less that the lower threshold for a given frame, the participant was classified as gazing left. If the value lay between the upper and lower thresholds, the participant was classified as gazing ahead, or to the ‘centre’. And if the value exceeded the upper threshold, the participant was classified as gazing right. The procedure for converting these ordinal conversions to annotations following the precedent set by existing manual gaze annotation is described in §III-C.

3) Head Posture Analysis: Head posture analysis enables a rotation vector to be calculated that indicates the direction a participant is facing. The solvePNP function of OpenCV calculates the relationship between points on a 2D image representation and their positions in a model of 3D space. In this case, the two sets of points used were the facial landmarks located in the image using Dlib, and the positions of these same landmarks on a hypothetical 3D model of a human head, where the origin (0,0,0) is found at the tip of the nose.

The landmarks used in this case were the chin tip, the nose tip, the two corners of the mouth and the outer corner of each eye. The correspondence between these sets of points was then calculated using solvePNP, which produces a rotation vector and translation vector. The data produced in this stage of analysis was not used in the creation of gaze annotations, nor in statistical analysis to establish a relationship between gaze behaviour and conversational dominance as the scale of the work required to complete other portions of this research was already extensive. However, it will hopefully serve as a useful addition to the MULTISIMO corpus for future work.

The results of head posture analysis are perhaps more useful when the resulting rotation vectors are converted to Euler angles.

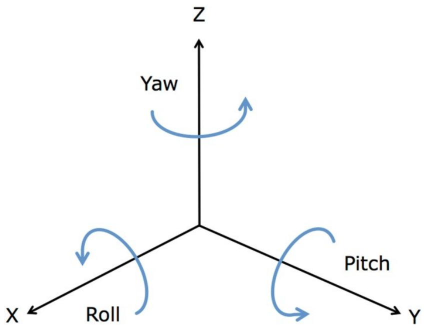

Fig. 5. The Euler angles

Yaw indicates the left-right turn of the head. Provided with some labelled training data, a model could be developed that classifies head posture into left, centre and right categories in the same manner as gaze direction analysis, which may be used to augment the accuracy of this previous stage or, alternatively, to provide an additional dimension of data for analysis.

C. Annotation

The previous three steps produce a data frame containing a row for every frame of each video with a column each for the relevant index, ordinal direction, rotation vector and timestamp (to give an idea of scale, there are 18,000 frames in a 10 minute video and 3 videos to be analysed per session). A filtering and merging process is necessary for producing useful annotations from this. Each annotation in ELAN must have a start time, an end time, a duration and an annotation value. Timestamps obtained directly from the source video by OpenCV were added to the dataframe.

As previously mentioned, the intention of this step was to produce annotations that reflect the format of human gaze annotation as faithfully as possible, with the additional annotation value of 'blinking' for one of the types of annotation produced. The first step of the process was to convert ordinal direction values of 'left', 'centre' and 'right' to the desired annotation value. This annotation value is dependent on the position of the participant in question to the other two participants, and therefore the Python script generating annotations from the video analysis data had to be modified for each video to ensure each ordinal direction was mapped to the correct annotation value. For example, in the case of a player found to the left of the facilitator, a gaze direction of 'left' should be mapped to GAZE-FACILITATOR while 'right' should be mapped to GAZE_PLAYER. For a player found to the right of the facilitator, these annotation values should be switched. And for the facilitator, annotation values of GAZE_PLAYER-LEFT and GAZE_PLAYER-RIGHT are used for the left and right ordinal directions respectively.

Sequences of frames where the annotation value was the same were merged into a single annotation, with the start time being the timestamp for the first frame in this sequence, the end time being the timestamp for the last, and the duration being calculated by subtracting the former value from the latter.

1) Types & Filtering: One of the issues faced with video analysis is a lack of continuous, contextual reasoning across a series of frames. Our eyes may be fairly easily deceived momentarily, but our brains are generally quick to make inferences about reality based on 'commonsense' knowledge. A computer, by contrast, has no such domain knowledge – including no sense of what behaviours are normal – and therefore lacks an understanding that it is unlikely that a person went from gazing left to gazing right and back again in the span of 33 ms. Rather than trying to implement an algorithm that accounts for the results of preceding and succeeding frames at the image processing stage – which would be prohibitively difficult and technical for a project of this scope – it was found that some simple heuristics bring the performance of the system to a level sufficient for our needs.

Two versions of annotation were created for each video. The first fine-grained annotation is a closer representative of the actual output of video analysis. All entries with a duration of 100ms or less (approximately 3 frames of footage) were removed on the basis that there was a reasonably high probability that these could be misclassifications and, even if it were possible to filter out these misclassifications, such fleeting annotations were unnecessary for the purposes of this paper. Entries where the value was BLINKING were maintained regardless of brevity, as blinking is a very rapid process. Secondly, broad-grained annotations were produced at a later stage that are a better approximation of the type of annotation a human annotator might provide. Entries where value was BLINKING were removed, and neighbouring entries extended in order to fill the resulting gaps. This step was necessary both for the assessment of inter-annotator agreement – as the lack of annotation for blinking in the human annotations would make comparison difficult – and for the purpose of the analysis in subsequent research, as blinking behaviour disrupts the continuity of gaze annotation in a way that is incompatible with the metrics used. This blinking annotation may be transferred to another tier with ELAN.

2) Importing into ELAN: The CSV files generated by the annotation phase can be viewed in ELAN using the built-in functionality for importing this file type. Then it is relatively simple to add the relevant video source as a linked file and save the project as an EAF file, enabling all research and analysis to be performed in the same manner as with human annotations.

3) Inter-Annotator Agreement: The reliability of annotation was assessed by comparison with a human annotation for the same session. Note that granularity difference resulting from frame-by-frame analysis – a level of detail that is not be feasible in manual annotation – means that it is not a gold-standard but rather an independent alternative for comparison.

ELAN offers a facility for assessment of inter-annotator agreement between two EAF files. In this case, a modified Kappa statistic with a minimum of 60% overlap required was used. Raw agreement is insufficient for assessing reliability. If, for example, there are two possible annotation values and one occurs more frequently than the other in ground truth, an annotator could simply provide this more common value for all annotations and achieve a relatively high raw agreement score. By contrast, Cohen's Kappa is a chance-corrected agreement index that normalizes the observed agreement by the amount that could be expected by chance alone [22]. Even for humans, separating something continuous like the angle of gaze into discrete categories – such as left, centre and right – is difficult, with the exact point where one category meets another varying between annotators, so perfect agreement was not expected.

IV. Evaluation

Evaluation is considered with respect to corpus coverage (§IV-A) and annotator agreement (§IV-B).

A. Corpus coverage

Inadequate performance on 12 video files excluded 11 sessions from close evaluation. A session had to be entirely disregarded in the context of this research if even 1 of the 3 videos posed an issue for automatic annotation. However, automatic annotation was performed successfully for 42 out of 54 total video files, leading to an overall coverage of approximately 78% of the corpus, if only 39% of the sessions. Misclassifications impacting performance may broadly be organised into two categories. The first encompasses those resulting from true failures of the system to adequately locate the iris position. Causes included glare on the glasses of a participant – which prevented proper thresholding of the eye region – occlusion of the eye region by the frames of glasses, and deep shadows cast over the eye region caused by overhead lighting and bowed head posture. The second source of misclassification results from the relative angles and positions of participants and cameras (see Fig. 6). In a number of sessions a camera and/or the chair of a player moved, meaning that, although the system was adequately able to locate the iris and therefore determine the direction of gaze, problems arose in the conversion of ordinal directions to annotation values. As usual, frames where the iris was found in a central location were annotated as GAZE_AWAY, while frames where the iris occupied the left side of the pupil were annotated as GAZE_PLAYER (where the player being analysed was found to the right of the facilitator). However, due to the closer proximity of the camera (filming the player being analysed) and the other player, the movement of the iris between a central position – which ought to be annotated as GAZE_AWAY – and a central position – which ought to be annotated as GAZE_PLAYER – was not large enough to be detected. This resulted in many misleading annotations.

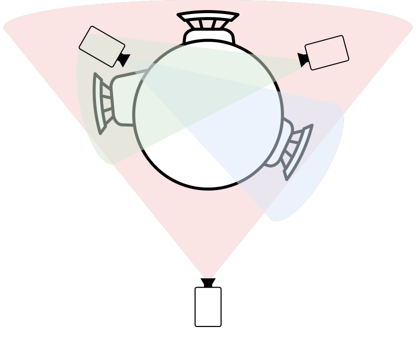

Fig. 6. A problematic room setup

The annotations for the remaining 7 of the 18 sessions were subjected to close evaluation. Three separate video files had to be annotated for each session (one per participant) and sessions lasted 10 minutes on average, meaning the following statistical analysis is based on approximately 210 minutes of video footage. The execution of the video analysis was observed for all 54 video files, and these 7 sessions were chosen on the basis of the assessment that classification of gaze direction was of at least as high a quality as those that would be produced by a human annotator.

B. Annotation Quality

Inter-annotator reliability was used to measure the agreement between automatically generated annotations for one session and the corresponding manual annotations. Manual annotations were also available for a second session, however, this session had to be excluded from analysis due to lighting conditions impairing performance for one of the participants: shadows on the eye region created by overhead lighting meant that establishing a threshold value that could adequately separate the iris from the sclera in a binary image was not possible, preventing accurate gaze annotation.

As mentioned in §III-C3, Cohen's Kappa is a metric that normalises the observed agreement by the probability of chance agreement, with 0 indicating that the two annotations in question are in total disagreement, while 1 indicates total agreement. A variant of Cohen's Kappa requiring a minimum of 60% overlap for a match to occur was used.

Participant	**Kappa**	**Raw Agreement**
P006	0.8416	0.9028
P007	0.6629	0.7841
Facilitator	0.875	0.9338
Average	*0.7932*	*0.8736*

TABLE I. MODIFIED KAPPA RESULTS FOR SESSION 2

The results of inter-annotator agreement assessment excluding unlinked values are presented in Table I. The average Cohen's Kappa for the three participants in Session 2 is 0.79, with a raw agreement value of 87%. Some researchers suggest that Kappa value of from 0.6 to 0.79 indicates substantial agreement while 0.8 to 1 indicates almost perfect agreement. Our value is found precisely on the cusp of the two. Others, however, are more demanding, however, with a Kappa value of 0.8 or higher providing good reliability, while a value of 0.67 to 0.79 allows tentative conclusions to be drawn [23].

V. Discussion

Annotation coverage of 78% the corpus provides a basis for future research. Manual annotation of gaze is an extraordinarily labour-intensive process, taking approximately 20 minutes to annotate 1 minute of footage for the MULTISIMO corpus [19], i.e. it would have taken approximately 70 hours to annotate the 210 minutes of video footage whose annotation is used for statistical analysis in this research. By contrast, it takes approximately 2.5 minutes to provide annotation for 1 minute of footage using this automatic system. This enables large quantities of data to be produced quickly and reliably, with only light supervision required at the beginning of analysis of each file to ensure environmental factors are not preventing accurate classification, thereby speeding annotation.

The quality of annotation and therefore utility of an automatic rule-based system such as this is contingent on assumptions about the environment, and its robustness is challenged in adverse conditions (poor lighting, inconsistent positioning of chairs and cameras, glare produced by the reflection of light on some participants' glasses). It is possible that 100% coverage is achieved with automatic gaze annotation in a corpus developed with consideration of this system's constraints. While the resulting data is evidently not as precise as that produced by e.g. gaze-tracking eyewear, it does allow rapid and low cost annotation (both in terms of labour and computational resources) without the need for specialist equipment.

In relation to the Kappa measure of 0.79, while this is only based on one session, it provides validation that the system worked correctly in this case and suggests that the performance in other 6 sessions would be similar, given that these were hand-picked based on observed accuracy. There was a fairly significant number of unlinked annotations between the manual and automatic versions as 633 were annotations provided across the three participants in the manually annotated version versus 779 annotations by the automatic annotator (representing a 23% increase). This granularity difference will almost certainly have suppressed the Kappa statistic to a degree, meaning accuracy could indeed be even higher than measured.

VI. Conclusion

Performance of the automatic annotator could perhaps be improved by incorporating the head posture data generated, either by using it as a factor contributing to the process of classifying gaze direction or to flag potentially inaccurate classifications where a contradiction occurs. Alternatively, as gaze and head posture are two factors that are related but by no means dependent, head posture data could provide an additional tier of data for analysis.

Acknowledgment

The research leading to these results has received funding from the ADAPT Centre (Grant 13/RC/2106), the European Regional Development Fund, the EU H2020 programme under the Marie Sklodowska-Curie grant No 701621, and was supported by the GEHM research network (Independent Research Fund Denmark grant 9055-00004B).

References

[1] P. Baranyi and A. Csapo, "Cognitive infocommunications: Coginfocom," in *2010 11th International Symposium on Computational Intelligence and Informatics (CINTI)*, 2010, pp. 141–146.

[2] ——, "Definition and synergies of cognitive infocommunications," *Acta Polytechnica Hungarica*, vol. 9, no. 1, pp. 67–83, 2012. [Online]. Available: https://uni-obuda.hu/journal/Baranyi_Csapo_33.pdf

[3] P. Baranyi, A. Csapo, and P. Varlaki, "An overview of research trends in CogInfoCom," in *IEEE 18th International Conference on Intelligent Engineering Systems*, ser. INES, 2014, pp. 181–186.

[4] C. Vogel and A. Esposito, "Linguistic and behavior interaction analysis within cognitive infocommunications," in *10th IEEE Conference on Cognitive Infocommunications*, 2019, pp. 47–52.

[5] F. D'Errico and I. Poggi, "The parody of politicians," in *2013 IEEE 4th International Conference on Cognitive Infocommunications (CogInfoCom)*, 2013, pp. 423–428.

[6] C. Navarretta, "Predicting an individual's gestures from the interlocutor's co-occurring gestures and related speech," in *2016 7th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, 2016, pp. 233–238.

[7] K. Jokinen and S. Scherer, "Embodied communicative activity in cooperative conversational interactions - studies in visual interaction management," *Special Issue of Acta Polytechnica Hungarica: CogInfoCom 2011*, vol. 9, no. 1, pp. 19–40, 2012.

[8] J. Reverdy and C. Vogel, "Measuring synchrony in task-based dialogues," in *Proceedings of the 18th Annual Conference of the International Speech Communication Association (INTERSPEECH2017)*, 2017, pp. 1701–1705.

[9] M. Koutsombogera and C. Vogel, "Speech pause patterns in collaborative dialogs," in *Innovations in Big Data Mining and Embedded Knowledge*, A. Esposito, A. M. Esposito, and L. C. Jain, Eds. Cham, Switzerland: Springer, 2019, pp. 99–115.

[10] A. Kendon, "Some functions of gaze-direction in social interaction." *Acta Psychologica*, vol. 26, no. 1, pp. 22 – 63, 1967.

[11] M. Argyle and J. Dean, "Eye-contact, distance and affiliation." *Sociometry*, vol. 28, no. 3, pp. 289 – 304, 1965.

[12] D. Heylen, I. Van Es, A. Nijholt, and B. van Dijk, "Experimenting with the gaze of a conversational agent." in *Proceedings international CLASS workshop on natural, intelligent and effective interaction in multimodal dialogue systems*, 2002, pp. 93 – 100.

[13] C. H. Morimoto and M. R. Mimica, "Eye gaze tracking techniques for interactive applications," *Computer vision and image understanding*, vol. 98, no. 1, pp. 4–24, 2005.

[14] Y. Fukuhara and Y. Nakano, "Gaze and conversation dominance in multiparty interaction." in *2nd workshop on eye gaze in intelligent human machine interaction*, vol. 9, 2011, pp. 9 – 16.

[15] L. Chen, R. T. Rose, Y. Qiao, I. Kimbara, F. Parrill, H. Welji, T. X. Han, J. Tu, Z. Huang, M. Harper *et al.*, "Vace multimodal meeting corpus," in *International Workshop on Machine Learning for Multimodal Interaction*. Springer, 2005, pp. 40–51.

[16] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "Appearance-based gaze estimation in the wild," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4511 – 4520.

[17] M. Koutsombogera and C. Vogel, "Modeling Collaborative Multimodal Behavior in Group Dialogues: The MULTISIMO Corpus," in *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*, N. C. et al., Ed. Miyazaki, Japan: ELRA, May 7-12 2018, pp. 2946–2951.

[18] M. Tacchetti, *User Guide for ELAN Linguistic Annotator: Version 5.0.0*, The Language Archive, MPI for Psycholinguistics, Nijmegen, The Netherlands, December 2018.

[19] R. Costello, "Analysing dominance in multi-party dialogue," Bachelor's Thesis, Trinity College Dublin, 2018.

[20] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[21] K. Dawson-Howe, *A Practical Introduction to Computer Vision with OpenCV*. John Wiley & Sons, 2014.

[22] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and Psychological Measurement*, vol. 20, no. 1, pp. 37–46, 1960. [Online]. Available: https://doi.org/10.1177/001316446002000104

[23] R. Artstein and M. Poesio, "Inter-coder agreement for computational linguistics." *Computational Linguistics*, vol. 34, no. 4, pp. 555 – 596, 2008.