

## iHEARu-PLAY: introducing a game for crowdsourced data collection for affective computing

Simone Hantke, Florian Eyben, Tobias Appel, Björn Schuller

### Angaben zur Veröffentlichung / Publication details:

Hantke, Simone, Florian Eyben, Tobias Appel, and Björn Schuller. 2015. "iHEARu-PLAY: introducing a game for crowdsourced data collection for affective computing." In *2015 International Conference on Affective Computing and Intelligent Interaction (ACII)*, 21-24 September 2015, Xi'an, China, 891–97. Piscataway, NJ: IEEE.  
<https://doi.org/10.1109/acii.2015.7344680>.



# iHEARu-PLAY:

## Introducing a game for crowdsourced data collection for affective computing

Simone Hantke<sup>1,2</sup>

<sup>1</sup>Chair of Complex & Intelligent Systems,  
University of Passau, Germany  
Email: simone.hantke@uni-passau.de

Florian Eyben  
audEERING UG,  
Gilching, Germany

Tobias Appel

<sup>2</sup>Machine Intelligence & Signal Processing Group,  
Technische Universität München, Germany

Björn Schuller<sup>1,3</sup>

<sup>3</sup>Department of Computing,  
Imperial College London, UK

**Abstract**—We introduce iHEARu-PLAY, a web-based multi-player game for crowdsourced database collection and – most important – labelling. Existing databases (with speech and video content) can be added to the game and labelling tasks can be defined via a web-interface. The primary purpose of iHEARu-PLAY is multi-label, holistic annotation of multi-modal affective speech databases. Players perform labelling (or prompted recording) tasks and are rewarded with scores and prizes, which are computed based on the “correctness” of their annotations, e.g., the agreement with a pre-defined gold standard or with the other players. iHEARu-PLAY is implemented with the open source high-level Python Web framework Django and can be installed on Unix and Windows platforms. Its modular architecture allows for easy integration of custom extensions: New gaming components can be added as plugins in order to support new databases and modalities. Label categories for each database are individually selectable and editable. Audio, image and video annotation are currently supported. iHEARu-PLAY will be available to the research community as a ready-to-use web-service. Researchers can add their own databases, optionally post rewards, and receive annotation results in the end. General users can register to play the game, have fun, compete with other players, and at the same time support science.

**Keywords**— *crowdsourcing; gamification; data collection; annotation*

### I. INTRODUCTION

The success of supervised machine learning techniques for computational paralinguistics [1] classification (which includes tasks such as sentiment analysis and emotion recognition from the field of affective computing) depends highly on the quality of labelled training data. The manual annotation of data by an expert is the primary way of gathering labelled training data. This way of getting the labels can be very time-consuming and expensive [2], [3], [4]. Furthermore, the labels have to be estimated from the subjective opinion of a small

number of experts who can often disagree on the labels [2], [5]. Besides these issues with labels, most databases are small with respect to the number of subjects who participated in the data recording, often resulting in poor generalisation of the classifiers to new speakers.

This paper introduces an alternative way to gather annotated data by crowdsourcing non-professional subjects/annotators through a fun gaming platform. Crowdsourcing is an arising collaborative approach applicable among many other applications to the area of language and speech processing. To save costs compared to expert human annotation in labs, many projects have turned to crowdsourcing [6]. However, most crowdsourcing services rely on click-workers which are being paid a (rather low) compensation for their work efforts. The jobs performed by the click-workers are most often not very appealing and often they have no motivation to do the work, besides the poor monetary compensation they receive. This leads to poor results for some workers, increasing overall costs and reducing reliability.

We hereby introduce iHEARu-PLAY, an alternative crowdsourcing method which motivates people by giving them a fun and playful environment, where they can have fun and at the same time voluntarily help scientific research projects by annotating data. Instead of offering a financial incentive, people are primarily motivated to participate due to the joyful experiences of a game. Usually, the motivation for an individual to voluntarily contribute to a crowdsourcing project ranges from altruism, over ego, to a shared sense of purpose. But the pursuit of fun and enjoyment through games is also seen as an emerging trend [7]. On top of the intrinsic motivation of playing a game and helping science, various prizes and awards could be also given, e.g., for the best scores, the most frequent users, and/or for randomly selected winners.

This will further boost the users' motivation and clearly goes beyond the typical click-worker paradigm.

iHEARu-PLAY is an application where modular gaming components can be freely configured and new databases can be easily imported. Desired label categories for each database are individually selectable and editable. Even though iHEARu-PLAY's primarily intended area of use is labelling audio and speech databases for the iHEARu project<sup>1</sup>, it is modality independent, e.g., images and videos can also be loaded for annotation. A strong focus is put on usability, i.e., for the interface to be self-explanatory and as straight forward as possible. As a web-based system, it is accessible in all popular web-browsers on desktop PCs and mobile clients (e.g., smartphones or tablets).

In the following we give an overview on related tools in Section II. Then, Section III describes iHEARu-PLAY's concept and main idea, the gameplay itself is described in Section IV. Section V provides the gamification concept and the design principles. The technical realization is shown in Section VI, before summarising this overview paper in Section VII.

## II. RELATED TOOLKITS

In the past few years several crowdsourcing platforms have been established, examples include Mechanical Turk, CrowdFlower, Turkit, Mob4hire, uTest, Freelancer, eLance, Trada, 99design, Innocentive, CloudCrowd, and Cloud-Flower [8]. Further work in the field of games with a purpose is the ESP Game, which was presented by Luis von Ahn and his colleague Laura Dabbish in 2004 [9]. Back then, von Ahn already realized that the only way to obtain precise image descriptions is manual labelling, which is very costly. He had the idea of having people label these images, without them even realizing what they are doing, by playing a game. In 2007 the browser-based game ARTigo, which was greatly inspired by the ESP Game, was developed at the Ludwig-Maximilian University of Munich, where players have to label historical paintings [10]. As it was inspired by the ESP Game, the basic gameplay is very similar. The players are asked to describe the artwork with tags as accurately as possible while competing for points. Another related online toolkit is Foldit, originally developed as part of an experimental research project, and is currently developed by the University of Washington's Center for Game Science in collaboration with the UW Department of Biochemistry [11]. The objective of the game is to fold the structure of a specific protein as well as possible. As there are many possible structures, finding the best one is regarded as one of the hardest problems in biology today and current methods take a lot of money and time, even for computers [11]. By taking advantage of humans' puzzle-solving skills, scientists hope to find the best structures.

<sup>1</sup><http://www.ihearuu.eu/>

Players are competing with each in order to find the best solution. For each solution handed in, players receive a score, based on how well they did. The highest scoring solutions are then analyzed by the researchers.

## III. MAIN IDEA

Since games are a seductive method to encouraging people to participate in projects [12], our system for data annotation should behave like a game in order to motivate people to voluntarily sign up and annotate data. We like to motivate users and spark their interest, not just in using our platform once but rather to play on a regular basis. Instead of rewarding the users with money, the task should be intrinsically motivating thanks to the scientific purpose and the deployed game-concept. Last but not least, we like the players to have fun while playing since this will be an important intrinsic motivator due to the lack of extrinsic monetary compensation.

Summarized, iHEARu-PLAY will

- have people annotate data while playing the game,
- motivate people with the help of gamification,
- reward people with virtual goods, such as scores and badges, instead of money per clicks,
- optionally hold sweepstakes with non-virtual prizes.

## IV. GAMEPLAY

When a new user visits iHEARu-PLAY for the first time, (s)he will be able to access a demonstration that explains the idea behind iHEARu-PLAY and teaches the user how to interact with the system. After signing up at iHEARu-PLAY and verifying the Audio-CAPTCHA the player gets presented with a list of databases. Each database features a description for the database, the number of included audio files, and questions associated with that database as well as a current multiplier which is used in the points calculation, described later on. This can be used to promote certain databases. Figure 1 shows iHEARu-PLAY's web-interface presenting the list of databases to the user.

Moreover, each user can see his or her personal progress for each database. After choosing a database the user will be presented with a random audio file and question from that database. By default the audio file will start playback automatically, this can however be turned off, which will then require the user to manually start playback by clicking the play button. After playback of half of the audio file, a list of answers fades in from which the user can select one (or sometimes multiple). After selecting his or her answer a submit button will fade in, which, upon execution, immediately presents a feedback message (based on the players performance), including the earned points as well as the next audio file and question. Then, the whole process starts over again. If the user earned

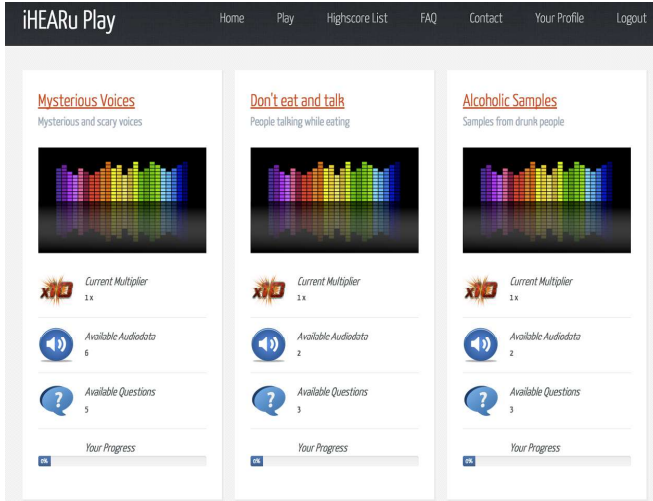


Fig. 1: Exemplary screenshot: iHEARu-PLAY features for each database a description of the presented audio files, the number of files, questions associated as well as a current multiplier.

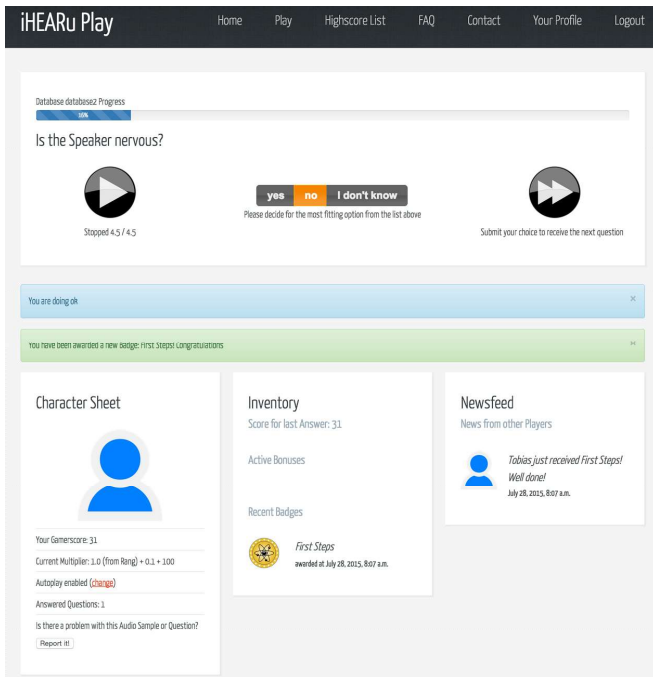


Fig. 2: Exemplary screenshot: iHEARu-PLAY's web-interface for labelling tasks, showing the progress for each database, list of answers, the feedback message after submitting the answer, including the earned points, followed by the next audio file and question.

a badge it will be displayed in the same area as the feedback message and (if the user allowed to share his or her activity on the platform) on the activity ticker, thus visible to all other players. Figure 2 shows the web-interface for labelling tasks, presenting the described progress for each database, the list of answers, the feedback message after submitting the answer, and the earned points.

Since each answer is independent from another, and the order of audio files is random, the user can always leave the game at any time without losing any progress. Thus the player is not forced to spend a fixed amount of time, but rather can decide for himself how many files to label in one session. Besides playing the game, the user can customize his or her profile, check his or her current badges and requirements for not yet unlocked badges or browse the leaderboards and other players' profiles.

## V. GAMIFICATION CONCEPT

As with traditional games, but also crowdsourcing systems, it is important to have a clear objective that is easy for the user to understand [13]. This is very important, as complex and explicit instructions do not only demotivate people, but also directly affect the resulting quality of the completed task [3]. If the challenge is too high, players will become frustrated but if it is too low, players will easily feel bored [14]. Objectives and goals have multiple characteristics that need to be relayed to the player: difficulty, distinctiveness, time perspective, scale and complexity [15]. Ideally, we always present a main goal the player has to reach, and in between, multiple intermediate goals as steppingstones while each goal is continuously challenging.

Further, interaction with the system will be - in the best case - self-explanatory and as straight forward as possible. Complicated rules or controls that need to be learned and memorized first, will result in a steep learning curve and can turn people away from the game [16].

Moreover, feedback needs to be immediate, otherwise the player does not know what effect his action caused and he will quickly become distracted and lose focus on the task at hand [14]. Variety and intensity of feedback is very important as it should also give information on how well the player did [17]. Instead of just selecting a random answer, users should focus and think before labelling any audio file, because a 'good' answer will yield a better reward. In order for this to work we need means to identity and distinguish between 'good', 'neutral', and 'bad' answers: Once the player submits the answer, it will be evaluated and scored. Depending on the type of question, different complex formulas are used to score the answer itself. Based on the resulting score of the answer it will be graded 'bad', 'neutral', or 'good'. The submitted answer is added to the list of answers before the evaluation. Thus, players answering a question that no one else has before, are not at a disadvantage as it will result in a good or neutral grading. We will describe these formulas in detail for the three possible questions now:

- **Single-select questions:**  
The user has to select a single answer from a list of choices. After the user submitted his or her answer, the system will calculate the mode (the answer chosen by the

most players) and also calculates the distribution over all choices. The evaluation involves three steps, independent from one another:

- 1) If the answer is equal to the mode, its score will increase by 3.
- 2) If the answer received more than 50% of the total votes from all choices, its score will be increase by 3.
- 3) If the answer received more votes (in percent) as expected by a uniform distribution (e.g. for 4 possible choices, the answers total percent have to be greater than 25%) its score will increase by 1.

A final score in the range of [1, 3] will result in a neutral grading, lower and higher will be bad and good, respectively.

- **Multiple-select questions:**

The user can choose multiple answers from a list of choices, however at least one item has to be selected with a maximum of all available choices. Since the total number of votes over all possible items, can be much higher than the number of players that answered that question, we will have to calculate two different percentage values for each item. We call the percentage of players that selected this item the 'global percentage' and the number of votes received, relative to the other choices, the 'local percentage'. Each selected choice is then evaluated in two independent steps:

- 1) If the local percentage is higher then expected in an uniform distribution its score is increased by 1, if it is lower or equal its score will decrease by 1.
- 2) If the global percentage is higher than 50% (that means, more than half of the players selected this item as part of their answer), its score will increase by 3, otherwise it will decrease by 3.

The same two steps will be applied to all items not selected by the user, however the scoring will be different as before. If the final score is equal to 0 it is considered a neutral answer. A positive value is considered good and a negative one bad, respectively.

- **Likert-Scala questions:**

The user has to select an integer value from a 5-point Likert-Scala, although other values (including negative numbers) are supported by the system as well. For each question the median, mean and mode will be calculated from all previously handed in answers. The evaluation involves three steps:

- 1) If the answer is equal to the median its score will increase by 2.
- 2) If the answer is equal to the (rounded) average, its score will increase by 1.
- 3) If the answer is equal to the mode, its score will increase by 3.

A score in the range of [1, 3] is regarded neutral, higher or lower values good or bad, respectively.

Before a player can receive the points based on his answer, the graded answer will be evaluated against the other modules of our gamification concept.

**Rank System:** Each player starts with the rank 'Beginner'. Upon reaching a certain amount of points players will advance to the rank 'Expert', which will earn them the ability to access other players profiles, view new leaderboards and earn points even while being offline. However the aim for each player should be to obtain the currently highest rank of 'Master' which will be awarded upon reaching even more points. At rank Master, players have the same rules and permissions as the rank of Expert, except the Master leaderboards which are of course exclusive to players with the rank of Master.

**Leaderboards:** iHEARu-PLAY features several leaderboards based on the following criteria: amount of points accumulated, number of questions answered, databases completed and number of badges awarded. Additionally each of these leaderboards will be available for the individual ranks (Beginner, Expert, and Master rank), resulting in twelve leaderboards. Only the leaderboards for the current rank of the player will be visible though.

**Points and Multipliers:** As described, each answer given by the player is analyzed and rated good, neutral or bad. This will set the base points to 20, 10 or 5. The final points awarded to the player are based on the given formula:

$$points = base(M_r + M_c + \frac{T_u}{100} + M_d),$$

where: *base*: base points,  $M_r$ : rank multiplier,  $M_c$ : users' current multiplier,  $T_u$ : users' trustability score,  $M_d$ : databases' multiplier.

Points have to be calculated with floating points numbers but will be rounded to an integer value before assigned to the player. The multipliers are based on the current rank and can further be adjusted through bonus items. Additionally, the users current multiplier will reset every day at midnight and will increase by 0.1 for the first ten answers given on the current day to a maximum of 1.0 (excluding modifications from other bonuses).

**Points from other Players:** Players with the rank of Expert or higher can earn points through the actions of other players. If you are the first Expert to chose a unique answer to a question, the system will make a special note of it. If other Expert players chose the same answer, thus reinforcing your decision, you will receive ten bonus points each time. These points will even accumulate while the original Expert is offline and will then be assigned to his or her user profile on this next login. (S)He will not be able to see from which individual players (s)he received those points though in order to suppress collusion between players.

**User Trustability:** Each user starts with a trustability of 100%. Single- and multiple-select questions can contain

TABLE I: SOME POSSIBLE TYPES OF BADGES AND THEIR CONDITIONS

Badge Name	Conditions
Early Bird	Answer 100 questions between 4 am and 7 am
Night Owl	Answer 100 questions between 23 pm and 2 am
Expert	Reach a score of 5000 Points
Master	Reach a score of 20000 Points
Powerman	Collect 100 Bonus Items (in total)
Regular Customer	Have a constant log-in streak of 7 days in a row
Way to go	Answer 100 questions in total
Autobiographer	Fill out own bibliography
Chatterbox (hidden)	Used the contact form 5 times

control answers which will result in a 50% decrease the users trustability if (s)he selects any of these. If the user does answer a control question correctly, his or her trustability will increase by 10%. Trustability does affect points awarded to the user. At the same time we track the amount of wrong control answers and can use this value to detect players who do not read the question and just select a random answer.

**Bonus Items:** Every time the player submits an answer, (s)he has a chance to find a bonus item. The probability starts at 1% and increase by another 1% for every answer submitted until a bonus item has been awarded. After this it will reset to the base of 1%. Currently implemented bonus items contain bonus points, modifier modifications and bonus probability modifiers. Upon rewarding a bonus item, one item is randomly selected from a list of all available bonus items, based on its own probability value. Using a weighted random sampling algorithm [18] we were able to give each bonus item its own probability value. Aside from bonus points, other bonus items have a dynamic durability and are active only a certain count of questions. Multiple bonuses can be active at the same time though.

**Badges:** Badges are awarded to players who fulfill special requirements unique to each badge. Examples include number of questions answered, number of times logged in, using the platform at special times (e.g. during the night) or anything else we like to come up with. Every time the player interacts with iHEARu-PLAY, either by logging in or by submitting an answer, all not yet obtained badges are checked to verify if the requirements have been fulfilled. Players are able to view the profile of other players and see their acquired badges. The list seen in Table I serves as an basic example for possible types of badges.

**News Ticker:** Recent achievements from other players will be visible in a live updating part of the interface. These can include badges awarded, rank up, bonus items found, or reaching #1 on one of the leaderboards. Each player can select to keep this information hidden. In this case his or her achievements will not be shown on the news ticker and also his or her profile will be inaccessible for other players.

## VI. REQUIREMENTS AND TECHNICAL REALIZATION

Additional functional requirements and constraints were taken into consideration when designing the gamification concept:

- iHEARu-PLAY will make use of HTML 5 in order to run on any modern device/browser without utilizing additional plug-ins (e.g. Flash)
- Countermeasures to discourage and detect malicious users and bots are applied
- Users require to pass an Audio-CAPTCHA before playing the game in order to determine if the user is able to play back and understand audio samples but also to filter out bots
- For an automatic quality control, each user has a trustability value, that is associated with his profile and thus all of his or her answers
- Administrators have the ability to detect and remove ‘bad’ answers or even single users from the system
- The system is completely agnostic to the content of the audio recordings

For the technical realization, the free and open-source high-level Python Web framework Django [19] was chosen. It comes with much built in functionality and can further be extended through 3rd party modules, called apps. Since it runs on Python, Django is essentially platform independent and can be deployed to Linux, Windows or OSX servers. Thanks to its template engine, the layout and design of the web application can quickly be replaced and also allows easy localization. The current prototype uses a free HTML 5 theme which is also responsive and thus will display correctly on a mobile device with a reduced screen size as well. Unregistered users are able to access a demonstration site that explains the idea behind iHEARu-PLAY and how to interact with the system. Users can then sign up with our platform, either by creating a local account or by using a 3rd party OAuth provider (e.g. Facebook). Although the player is never asked for his or her name (and this information is also not pulled from Facebook or any other OAuth provider) the system has to store the email address and other personal information, like age, gender, mother language, English proficiency, and speaker setup, necessary for participating of each player. The email address is used as a unique feature to identify different players, and has to be validated before playing the game. Also the user has to pass a one-time Audio-CAPTCHA, consisting of a random four digit number. Once the player selects an available database, the browser will start playing back a random audio file from that database. For the playback, the default HTML 5 audio controls are used, however the controls are hidden so that the user can not seek during playback. A toggle button for pausing and playing is available to the user instead. Using jQuery and Javascript we are able to hide the list of possible choices until the audio file has played past the half of its total

length. This is to thwart bots and randomly clicking players by forcing them to listen to at least the first half of the audio recording. Only after the user selected an answer the submit button will be visible. All forms are automatically protected from Cross-site-request-forgery thanks to Django's csrf token. With each log in, the user profile is updated and the last login value is stored. This is important, as it is used to detect a streak of daily logins. Every time the user profile is saved, either by login in or out, or by receiving points after submitting an answer, all available, not yet obtained, badge conditions are checked against that user profile. Depending on the amount and complexity of badges this can become inefficient, however it has certain advantages. The current implementation allows for nearly unlimited badge conditions to be created, without having to rely on a complex and heavyweight rule engine. As Django uses an object-relational mapping (ORM), all database tables and attributes are defined as Python classes. This allows rapid development, and changes to the current database model, without losing data. When an answer is submitted we do not only save the answer tuple (consisting of user : userID, question : questionID, audioData : audioDataID, timestamp : DateField, body : string), but also update the total count for this answer in a separate table. This allows quick look-ups on the count of answers for a specific question, which is needed to evaluate the rewarding grade for a new submitted answer. The system expects the audio files to be stored somehow (e.g. FTP) on the local (or net mounted) hard-drive. When the administrator wants to add a new database, (s)he will just create it in the admin interface and point it to the local path where the audio files are being stored. iHEARu-PLAY will automatically search the given path for all files and create audioData entries in the database. In a second step, it will create entries in the AudioQuestPair table based on the questions enabled for this database by the administrator. Thus it will create an entry for every possible combination of audio data and question.

## VII. CONCLUSION AND OUTLOOK

We have introduced iHEARu-PLAY, a modular, cross platform, browser-based crowdsourcing game implemented with the high-level Python web-service framework Django. iHEARu-PLAY is an application where modular gaming components can be freely configured and new databases can be easily imported. Categories for each database are individually selectable and editable to allow for maximum customization. Even though iHEARu-PLAY's primarily intended area of use is labelling audio databases, it is principally modality independent, e.g., images and videos could also be imported. A strong focus was put on making the interaction with the system as easy as possible, i.e., a self-explanatory interface and game rules. Due to its browser-based design, the system runs on all major device platforms, such as Windows, Unix, Mac and even Mobile Devices.

The development of iHEARu-PLAY is still active and due to iHEARu-PLAY's modular architecture, rapid addition of even more new and diverse features is possible and planned.

Future work will focus on improving the quality and abuse management. One security feature could be tracking the players choice selection by position in the interface. I.e., if some user always selects the choice on the far right, for example, this possibly indicates that (s)he is not clearly thinking about the answer.

Furthermore, gamification elements will be expanded, for example adding additional ranks and thus gradually giving users more rights or access to other features of the system. Moreover, a customizable avatar could be implemented. A lot of games and gamification enhanced applications, give the player the option to design and customize his or her own avatar which (s)he can present to other players as well. Combined with new bonus items, this would allow players to show off their achievements in a completely new way.

We are planning to integrate an option to allow users to post achievements in their social networks, e.g., Facebook Walls, Google+, or other social media sites – of course just if the user wants to. At present, only the authentication of users via third party authentication providers, such as Facebook, is implemented.

Another planned extension will be a speech recording tool, which allows to collect an induced or prompted database of audio and video from the user – of course only given the user's explicit consent. Microphones and cameras which are embedded in most laptop PC, tablets, and smartphone devices can be used to do the recordings. With this feature, collection of speech data under real-life conditions in the wild (microphone/webcam type, devices, background noise, reverberation, light adjustment etc.) of a large number of subjects with different geographic origins, languages, dialects, cultural backgrounds, age groups, etc. will be possible.

Thus, iHEARu-PLAY has a great potential of creating and driving community supported, large-scale, real-life data-collections for advanced paralinguistic tasks. This could allow for fields such as multi-modal sentiment analysis to make a jump to the next level of performance, such as in the field of automatic speech recognition with the large data collections by the big, industry leading companies.

## ACKNOWLEDGMENT

The research leading to these results has received funding from the European Community's Seventh Framework Programme under grant agreement No. 338164 (ERC Starting Grant iHEARu). We thank audEERING for technical support and hosting of the iHEARu-PLAY service.

## REFERENCES

- [1] B. Schuller and A. Batliner, *Computational Paralinguistics: Emotion, Affect and Personality in Speech and Language Processing*. Wiley, Nov. 2013, 344 pages.
- [2] A. Tarasov, S. J. Delany, and C. Cullen, "Using crowdsourcing for labelling emotional speech assets," 2010.
- [3] V. Ambati, S. Vogel, and J. G. Carbonell, "Active learning and crowdsourcing for machine translation," in *LREC*, vol. 1. Citeseer, 2010, p. 2.
- [4] A. Kittur, E. Chi, and B. Suh, "Crowdsourcing for usability: Using micro-task markets for rapid, remote, and low-cost user measurements," *Proc. CHI 2008*, 2008.
- [5] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy, "Learning from crowds," *The Journal of Machine Learning Research*, vol. 11, pp. 1297–1322, 2010.
- [6] H. Zhai, T. Lingren, L. Deleger, Q. Li, M. Kaiser, L. Stoutenborough, and I. Solti, "Web 2.0-based crowdsourcing for high-quality gold standard development in clinical natural language processing," *J Med Internet Res*, vol. 15, no. 4, p. e73, Apr 2013. [Online]. Available: <http://www.jmir.org/2013/4/e73/>
- [7] B. M. Good and A. I. Su, "Games with a scientific purpose," *Genome Biol*, vol. 12, no. 12, p. 135, 2011.
- [8] A. Doan, R. Ramakrishnan, and A. Y. Halevy, "Crowdsourcing systems on the world-wide web," *Commun. ACM*, vol. 54, no. 4, pp. 86–96, Apr. 2011. [Online]. Available: <http://doi.acm.org/10.1145/1924421.1924442>
- [9] L. von Ahn and L. Dabbish, "Labeling images with a computer game," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '04. New York, NY, USA: ACM, 2004, pp. 319–326. [Online]. Available: <http://doi.acm.org/10.1145/985692.985733>
- [10] L.-M. U. of Munich, "Artigo - social image tagging," 2007. [Online]. Available: <https://www.artigo.org/about.html>
- [11] C. for Game Science at University of Washington, "foldit," 2015. [Online]. Available: <https://fold.it/portal/info/about>
- [12] L. Von Ahn, "Games with a purpose," *Computer*, vol. 39, no. 6, pp. 92–94, 2006.
- [13] E. Estellés-Arolas and F. González-Ladrón-de Guevara, "Towards an integrated crowdsourcing definition," *Journal of Information science*, vol. 38, no. 2, pp. 189–200, 2012.
- [14] J. Schell, *The Art of Game Design: A book of lenses*. CRC Press, 2014.
- [15] U. Kleinbeck, "Handlungsziele," in *Motivation und Handeln*. Springer, 2006, pp. 255–276.
- [16] A. K. Przybylski, C. S. Rigby, and R. M. Ryan, "A motivational model of video game engagement," *Review of general psychology*, vol. 14, no. 2, p. 154, 2010.
- [17] J. McGonigal, *Reality is broken: Why games make us better and how they can change the world*. Penguin, 2011.
- [18] P. Efraimidis and P. Spirakis, "Weighted random sampling," in *Encyclopedia of Algorithms*, M.-Y. Kao, Ed. Springer US, 2008, pp. 1024–1027. [Online]. Available: [http://dx.doi.org/10.1007/978-0-387-30162-4\\_478](http://dx.doi.org/10.1007/978-0-387-30162-4_478)
- [19] "Django - computer software," 2014 (Version 1.7). [Online]. Available: <https://djangoproject.com>