Adaptive questionnaire design using AI agents for people profiling

Keywords: AI, classification, behaviors, profiling

Abstract: Creating employee questionnaires, surveys or evaluation forms for people to understand various aspects such as motivation, improvement opportunities, satisfaction, or even potential cybersecurity risks is a common practice within organizations. These surveys are usually not tailored to the individual and have a set of predetermined questions and answers. The objective of this paper is to design AI agents that are flexible and adaptable in choosing the survey content for each individual according to their personality. The developed framework is open source, generic and can be adapted to many use cases. For the evaluation, we present a real-world use case of detecting potentially inappropriate behavior in the workplace. In this case, the AI agents that create the personalized surveys act similarly to a human recruiter. The results obtained are promising and suggest that the decision algorithms for content selection approaches are similar to a real human resource manager in our use case.

1 INTRODUCTION

The main idea of this work is to replace some of the manual work of professionals in surveying people in large organizations with AI agents that are able to perform the survey process and automatically mark the answers. AI is used to create dynamic surveys that ask questions from a pre-built data set of questions that hide inherent attributes, ambiguities, and importance factors. The disadvantage of creating a fixed set of questions (which must also be a small number, otherwise respondents may not focus on the sequence of questions) is that at the end of the survey, only a text-based or numerical representation with the same categories of questions asked for everyone is available for further offline evaluation of individuals. The task of the AI system is to identify the sequence of questions to be asked in order to better score individuals on the survey objective with the same number of questions as a typical fixed survey.

The contributions of our work are the following:

1. The first work that implements AI agents capable of assuming the role of a human expert to create dynamic, adaptive surveys to evaluate human profiles with respect to various goals.
2. An open-source framework that can be customized and extended by users to set up AI agents that can conduct surveys. The implementation, setup for a new user, and available internal AI algorithms are independent of the context, i.e., the characteristics used by the survey questions and their evaluation goal. It is available at this link https://dl.dropbox.com/scl/fi/mvqbqnqf9h581gs88svuc/ai-master.zip?rlkey=8odgdp1b9h0vl5p7yuko8tl0b containing our real, anonymized dataset and results.
3. Data science tools to support organizations after data collection and aggregation at individual and team levels are provided by the framework.
4. A novel abstraction of survey organization using questions, videos, and underlying hidden attributes to drive AI agents.

The rest of the article is organized as follows. The next section contains a literature review of published work and methodologies that have inspired our framework design, use cases, and methods. Section 3 describes a specific use case in industry that provided further insight into the generalizability of the methods to other use cases. Section 4 presents the implementation details behind the proposed AI agent and the data analysis tools used for the post-survey. The evaluation section provides comparative results of methods, previous experiments and observations. Finally, the last section presents the conclusion and plan for future work.

2 RELATED WORK

Profiling and clustering individuals using data mining and NLP methods to extract data from textual data is a common trend in the literature. In (Wibawa et al., 2022), the authors use AI techniques like classical NLP to process application documents for open positions and automatically filter, score, and prioritize candidates. This helps recruiters select the most

promising candidates within a limited resource budget. On the other hand, the work in (Bajpai et al., 2023) discusses methods to automatically create company profiles and clusters based on employee reviews on the employer rating platform Glassdoor[1]. The available reviews are subjected to aspect-based sentiment analysis, where the term *aspect* is used to organize the sentiment analysis embeddings according to specific groups of features such as salary, location, work life, and so on. Then, the features are clustered using machine learning to profile the targets by categories. The social media data extracted from WeChat [2] is used in (Ni et al., 2017) to build profiles of individuals and cluster them according to their field of activity, using NLP techniques such as those mentioned above. The work in (Schermer, 2011) discusses data mining used in automated profiling processes, with a focus on ethics and possible discrimination. Use cases such as security services or internal organizations that create profiles to evaluate various characteristics of their employees are mentioned. Profiling individuals for content recommendation, such as news recommendations, has been used for many years(Mannens et al., 2013). Automatic detection of fake profiles on social media platforms such as Instagram and Twitter is another widespread use case for people profiling using data mining and clustering techniques (Khaled et al., 2018). On the commercial applications side, we mention Relevance AI [3], which handles most of the above cases using data mining and clustering for industry.

Another interesting application of automated profile identification within an organization is presented in (Rafae and Erritali, 2023), which creates a recommender system that automatically suggests employees capable of performing various tasks within internal projects of an organization. The profile of each employee is created by analyzing data from the human resources department, emails, messages and publications sent by the employee, as well as previous tasks completed by the employee, including the correctness of the solution. The recommendation system then matches the goals of a project and a specific task with the profile to create an evaluation score. The questionnaire design is usually a tedious task requiring expertise (Lietz, 2010) and sometimes must be based on various types of standards such as Employee Screening Questionnaire (ESQ-2) (Iliescu et al., 2011). Industry also recently started using AI in questionnaire creation using chatGPT-like prompts - e.g. SurveyMonkey Genius[4], but they cannot be easily dynamically adapted for the person filling-in the questionnaire.

The automated techniques described above complement the techniques we propose. We believe that our methods are novel in that we profile an individual in real time using an adaptive technique that asks questions which lead to the classification of the individual into specific profile categories. Data mining and adapting the language depending on the answers can also be done in addition to our methods. This is the reason why we consider our work complementary to the literature above.

3 METHODOLOGY OF CREATING SURVEYS' FEATURES AND DATASETS

The idea for our work and the methods used stem from a specific use case used by vorteXplore[5] in a client company to identify potentially inappropriate behaviors of individuals and teams within the company. The *Inappropriate* workplace behavior is typically engaged in by a limited number of individuals, but other group members may be affected by, track, and even participate in it to varying degrees. Note that for reasons of confidentiality and general ethics of data science, we cannot disclose the name of the organization in which this specific survey was conducted. In the remainder of this section, we briefly outline the applied use case.

The surveys created in our applied use case are conducted at the individual level and used to collect data and predict inappropriate behavior (IB) within teams in a large organization. VorteXplore measures each employee's tolerance for inappropriate behavior and how likely they are to recognize and respond to that behavior. These results are compared to the organization's tolerance policies, and the extent to which existing sanctions, training, and other methods considered are likely to be effective. Post-survey analysis focuses on three areas: (a) the individual, (b) each individual's expectations of the team's response, and (c) the team's risk profile compared to that of the organization as a whole. A concrete IB situational awareness could be to assess the potential for bullying within an organization or team. It can be measured by prevalence and frequency. However, less severe but frequent incidents are more difficult to de-

[1]Glassdoor.com

[2]WeChat.com

[3]https://relevanceai.com/for-analytics-insights

[4]https://www.surveymonkey.com/mp/surveymonkey-genius

[5]https://vortexplore.com

tect but represent the most common form of bullying (Samuel Farley and Niven, 2023), (Staale Einarsen and Notelaers, 2009).

The methodology for putting the developed framework into practice can be outlined as follows:

- Work with each client to identify behaviors prevalent in the workplace and design a database of questions that not only help identify behavioral hotspots, but also provide high response rates, comparable results, and a high degree of validity.
- Triangulate areas of the organization where unhealthy behaviors are prevalent down to individual teams or small groups of individuals (usually 5-10) using data science tools.
- Identifies areas where group behaviors appear unhealthy and could lead to conflict within the team or infect other teams.

The recording of videos/images and questions are specific to the needs of each client and must be created manually by professionals. They are defined once and then reused. A library of attributes for each media file and question is included in the framework, but can be customized by clients.

4 METHODS

The first part of this section defines the abstractions needed for defining a database of questions, videos/images, attributes, and cluster categories, required for the survey process. As mentioned in Section 3, each client has to define these once in their organization such that their interest is accomplished at the end of the evaluation through the surveys performed by the AI agents. However, note that the design of the current framework version promotes shearability between these features, as a unified library that clients could just reuse out of the box. This could be valuable both from a community point of view, but also from an internal organization recurring surveys.

4.1 Surveys dataset creation

A survey is composed of several videos/images aka clips shown to a person. On each clip shown there will be a collection of questions based on a specified compatibility graph. In our design, the collection of abstract items and entities that exist in a survey dataset is described below:

- *Clips*. A collection of assets representing video files, messages, media posts, etc. Clip indices also have an optional dependency specification, i.e., the client can impose that a new clip should depend on an already shown set of other clips: $Dependencies(C_i) = \{C_j\}_{j \in 1..|C|}$.
- *Attributes*. A set describing the properties of each clip asset. Examples from our use case of IB recognition: *inappropriate touch*, *offensive language*, *leadership style*, etc. These are set by the organization creating the content and are not visible to the respondent. Intuitively, when a questionnaire record is created, the attributes are set by the client depending on what they are looking for.
- *Attributes*. A set describing the properties of each clip asset. Examples from our use case of IB recognition: *inappropriate touch*, *offensive language*, *leadership style*, etc. These are set by the organization creating the content and are not visible to the respondent. Intuitively, when a questionnaire record is created, the attributes are set by the client depending on what they are looking for. The vector of all attributes (ordered by index) is given in each clip, with values between 0-1 representing the importance of the attribute to the content of the clip, i.e. a value of 0 means there is no relationship between them, while a value of 1 represents an important correlation. Formally, a clip $C \in Clips$ has a set of $Attr(C)\{Attr_{C_1}, Attr_{C_2}, ..., Attr_{C_{NAttr}}\}$, where $NAttr|Attributes|$. Then, the importance of an attribute $A \in Attributes$ within a clip C is represented by $VAttr(A,C)$.
- *Categories*. A category of the clips being asked. Examples from our use-case: *Sensitivity*, *Awareness*.
- A collection of questions Q, where each $Q_i \in Q$ has the following properties:
  - The set of clips in which this question can be asked $Clips(Q_i)\{C_j \; inClips\}_j$.
  - The dependencies of this question. This takes the form of a directed acyclic graph where each Q_i has a set of dependency questions $Dependencies(Q_i)\{Q_k\}_k$, meaning that there is a hard constraint on asking one of the questions Q_k to allow the follow-up question Q_i.
  - Attributes behind the questions, similar to those in the clip definition above, i.e. $Attr(q)\{A \in |Attributes|\}$, and their numerical importance value in the respective question, $VAttr(A,q)$.
  - Severity (Q_i) - how important is the question overall.
  - Baseline(Q_i) - expected value of the question according to the culture of the organization (customer).

– Ambiguity(Q_i) - the ambiguity of the question. There is no reason to design an ambiguous question, but sometimes survey analysis shows that some of the questions may indeed be ambiguous, and this factor is used to mitigate responses when this is the case.

- Profiles or Clusters specification. The goal of the survey is to classify a person into a particular profile or cluster, as described later in the text. In this process, HR professionals must define attributes $Attr_i \in Attributes$ and categories $Cat_i \in Categories$ that they are interested in for cluster definition. The aggregation of these features is represented by equation 1. As shown in section 4.2, deviations from the baselines of the organization in the survey question are used along with these categories and attributes in computing the features required to converge to one cluster or another.

$$Feats = \{Cat_0, Cat_1, ..., Cat_{N-1}, Attr_0, Attr_1, ...Attr_{M-1}\} \quad (1)$$

When defining a cluster, each of these characteristics is specified as a Gaussian distribution with a median and standard deviation set by the organization. By denoting this set with *clusters*, Eq. (2) below shows such a specification of a single cluster using the mentioned features. The argument of using a multivariate Gaussian distribution to define each cluster using each feature deviation score (4.2 section) is that the organization can specify:

1. Intuitively, the median represents the value expected of a respondent with respect to that feature in order to place them in the target cluster.
2. The standard deviation, represents how tolerant they are for the deviation score of the feature and the target cluster.

$$\begin{aligned} Cluster_k = \{&(Cat_0, \mathcal{N}(\mu_{C_0}, \sigma^2_{C_0})), ..., \\ &(Cat_{N-1}, \mathcal{N}(\mu_{C_{N-1}}, \sigma^2_{C_{N-1}})) \\ &(Attr_0, \mathcal{N}(\mu_{A_0}, \sigma^2_{A_0})), ..., \\ &(Attr_{M-1}, \mathcal{N}(\mu_{A_{M-1}}, \sigma^2_{A_{M-1}}))\}, \\ &\forall k \in Clusters, \end{aligned} \quad (2)$$

Each of the *Severity*, *Baseline*, and *Ambiguity* properties have a numeric value between 1-7. Ambiguity can be null, i.e., not taken into account if so. The features of categories and attributes of a specific cluster k, as filtered from the Eq (1), are shown in Eq. (3).

$$Feats_k = \{Cat, Attr | Cat, Attr \in \cap(Feats, Cluster_k)\} \quad (3)$$

4.2 Features computed inside the AI engine

HR professionals typically seek to collect employee profiles and then generate statistics, both at the level of the entire company and by team and hierarchy, on topics of interest to them (in our use case, for example, the client wanted to understand IB's potential and its degree). The base function for profiling is *deviation*, which reports the answers of each question on the baselines of the organization. In the remainder of this text, we refer to P as the set of individuals in the organization and $team(P_i), P_i \in P$ as the numerical ID representing the team of an individual P_i.

4.2.1 Deviations

To calculate the raw deviation during and after the survey for each question, the response values and the organization's expected tolerance values (baseline values) (numeric values in the range 1-7) are compared. The deviation function is either linear or quadratic by default, as in equation (4), but can also be customized by the client.

$$Dev_{raw}(Q_i) = |Response(Q_i) - Baseline(Q_i)|^2 \quad (4)$$

For a better evaluation of the results, the ambiguity levels of the clips are also considered. This is used to lower the deviation of the question (linearly) if the client (organization) considers that the clip they selected has a certain level of ambiguity, Eq. (5).

$$Dev_{ambg}(Q_i) = \frac{Dev_{raw}(Q_i)}{Ambuiguity(Q_i)} \quad (5)$$

The severity level of the question also affects the deviation of one question by boosting (or scaling down) its original value. deviation, Eq. 6.

$$Dev_{final}(Q_i) = Dev_{ambg}(Q_i) \times Severity(Q_i) \quad (6)$$

4.2.2 Removing Anchor and constant over/underscoring the questions

In a survey, several biases can be observed in practice (Yan et al., 2018). Some of the most common

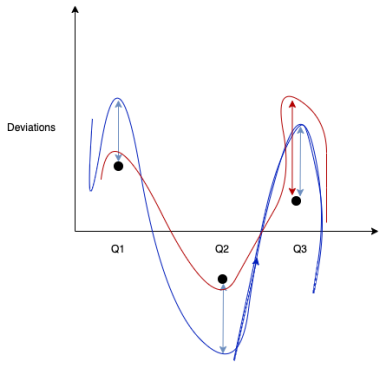

Figure 1: Two users (blue and red curves) answering the same three questions (Q1, Q2, Q3) in a survey. The black dots represent the average deviations for each of the questions (a 0 value deviation represents the baseline set by the organization). You can see that the blue user's answers deviate with an almost constant value in modulo, which means that the user may actually be overreacting. This means that it can be normalized, i.e. its average deviations are subtracted a little from the original response deviations. The red user, on the other hand, does not maintain the same constants for his deviations, e.g., the value for Q3 deviates greatly compared to the other two responses. This could be a strong signal that the user is very likely to answer Q3 without deviations and that the observed value is indeed correct.

are the anchors (attached to or influenced by a previously seen question) and the constant over- or underscoring of the answers. In order to obtain correct statistics at the team and organization level, the algorithm attempts to eliminate possible biases in a post-processing step, which may be more or less frequent depending on the personality. A concrete example explaining these biases and their removal is shown in Figure 1. Briefly, the method used is to try to find patterns in the deviance either throughout the survey or in short successive sequences (Dee, 2006).

Suppose user U_j answers the questions in his survey, denoted $QSurvey(U_j)$. $Dev_{final}(Q_i, U_j)$ denotes the deviation of user U_j on question Q_i in the survey, according to the baselines of the organization. Equation. (7) shows how the mean deviation values are calculated for a given user and survey. Note that the modulo operator is used because the deviations from the baseline can be either positive or negative numbers. Also, in practice, the formula is calculated for all consecutive sequences of questions, but for simplicity, it is presented calculated for all questions in the survey. The practical reason for this is to correctly capture anchoring bias, i.e., a question that has influenced the respondent to overreact over a short sequence of questions.

$$DevMeans(U_j) = \frac{\sum_{Q_i \in QSurvey(U_j)} |Dev_{final}(Q_i)|}{|QSurvey(U_j)|}, \tag{7}$$

Next, the mean and standard deviations of the user responses relative to the overall pattern in the organization can be calculated to obtain the constant deviation factor in the user responses. This is shown in equation (8).

$$ConstDev(U_j) = \frac{DevMeans(U_j)}{max(1, std(DevMeans(U_j)))} \tag{8}$$

Finally, the unbiased deviation of user response to a question can be adjusted then to the final form with bias removal, Eq. (9)

$$Dev_{final}^{NoBias}(U_j, Q_i) = Dev_{final}(U_j, Q_i) - sign(Dev_{final}(U_j, Q_i)) * ConstDev(U_j) \tag{9}$$

4.3 Scores feature vector

During a survey and post-survey analysis, the user $U(j)$ is characterized as a feature vector after each question: $Scores(U_j) = \{Scores_{U_j}^{Categories}, Scores_{U_j}^{Attributes}\}$. These scores can be calculated using the basic settings for variance values mentioned above. Suppose a user's survey U_j consists of a series of questions $QSurvey = \{Q_{id1}, Q_{id2}, ...Q_{id_N}$. Also remember that each question asked is part of a clip, $C_{id_i} \in Clips$. The equations (10), (11) show how these internal feature scores are calculated for each defined category of questions and attributes in the set. Note that the feature calculations of the scores and intuitively are a weighted average of the relevance of the attributes in the set of questions asked, and the same is true for the categories.

$$Scores_{U_j}^{Categories}[Cat] = \frac{\sum_{i=id_1}^{id_N} \mathbb{1}(Cat \in Q_{id_i}) * Dev_{final}^{NoBias}(U_j, Q_i)}{\sum_{i=id_1}^{id_N} \mathbb{1}(Cat \in Q_{id_i})} \tag{10}$$

$$Scores_{U_j}^{Attributes}[A] = \frac{\sum_{i=id_1}^{id_N} VAttr(A, C_{id_i}) * Dev_{final}^{NoBias}(U_j, Q_i)}{\sum_{i=id_1}^{id_N} \mathbb{1}(VAttr(A, C_{id_i}) > 0)} \tag{11}$$

4.4 AI driven agent

The purpose of an AI agent in this context is to create an adaptive dynamic survey to help classify as best as

possible an individual with a certain number of questions asked.
With the questions asked for a user U_j and responses obtained so far, at step i, i.e., after asking the $i'th$ question, from the AI agent's perspective the reviewed person state, Eq. (13), contains:

1. The sequence of questions already and the user responses, $QRset$.
2. The $Scores$ vector computed for the reviewed person so far using its deviations and responses.
3. A probability distribution of the user being part of each cluster is as follows.

   Knowing the answers and their deviations computed with $Scores^{U_j}$, the AI agent's internal state is a probability distribution over the clusters, i.e., $P(U_j \in Cluster_k)$, representing the probability of U_j being part of cluster index k. As shown in Eq. (12), this value is obtained by averaging the probabilities of the user response score deviations being part of each feature f's Gaussian distribution as initially specified by the organization in Eq. 2. Graphically, this probability is inversely proportional to the distance to the cluster, i.e. the closer is a user to a cluster, the higher the probability it has of being in that cluster.

$$P(U_j \in Cluster_k) = \frac{\sum_{f \in Feats_k} P(Scores_f^{U_j} \in \mathcal{N}(\mu_f, \sigma_f^2))}{|Feats_k|} \tag{12}$$

The state of the current review process can be formalized as in Eq. (13).

$$S_i = (QRset_i = \{Q_i, R_i\}, Scores_i, P(U_j \in Cluster_k)) \tag{13}$$

The algorithm starts with a 0-state containing equally split probabilities across clusters, without any prior assumption. Intuitively the AI agent must then choose at each step the next clips and question from the internal database to contribute in the end to the characterization of the reviewed person, i.e., classify as closely as possible according to the real profile of the person according to the specifications given by the organization. From a data science perspective, this is equivalent to eliminating the *entropy* (Aning and Przybyła-Kasperek, 2022) in the classification as best as possible within the limited number of questions in a survey process.

The method used in the algorithm for the decision-making part is similar to a contextual bandit problem in reinforcement learning (Park and Faradonbeh, 2022). The next action Act_i to take at each step by the agent corresponds to what pair of clips and follow-up questions to ask. This policy is denoted as $\pi(S_i, Act_i)$, where S_i is the embedded state of the agent (Eq. (13)). The action Act_i is also restricted by the constraints of the survey's flow and state regarding the set of previously asked questions and clips (Section 4.1).

At each step i, The decision-making algorithm has two main steps:

1. Choose the most promising cluster to test the agent against in the current state S_{i-1}.
2. Compute the most interesting clip or question, as constrained by the flow to ask in the selected cluster.

The second step is presented in Listing 1. Variables *prevClipId* and *prevQueId* represent the previous clip and question asked. *needAClip* is True when the next step is to display a new clip instead of a question, while *numQueInClips* contains the number of asked questions under the previously shown clip. Note also that a clip selection does not affect the step progress, Line 13. Also, the process of selecting the next clip c (Line 7), or a question q (Line 18) is a sampling process where the probabilities (of compatible clusters/clips after filtering by constraints) are obtained from a cosine similarity (Xia et al., 2015) of the attributes. This is sketched in the following text. In the first phase, the list of all compatible questions or clips filtered by the constraints of the survey's flow and cluster TC are gathered in $CompatibleSet$. In the second phase, Eq. (14), (15), for each of these questions or clips, all attributes in the dataset are gathered in separate feature vectors. Another similar feature vector is composed of the features given by the characterization of the target cluster, $Feats_{TC}$. In the third step, the cosine similarity applies between the pairs of feature vectors of the clips or questions mentioned above and the target cluster's interests. Finally, a probabilistic sample is drawn from the distribution obtained, Eq. (16).

$$\begin{aligned} VAttr(item_i) = \{&(A_i, VAttr(A_i, item_i)) \\ &| A_i \in Attributes \\ &\wedge item_i \in CompatibleSet\} \end{aligned} \tag{14}$$

$$VAttr(TC) = \{(A_i, VAttr(A_i, Cluster_{TC})) \tag{15}$$

$$\begin{aligned} &P_{selection}(item_i) = cosSim(VAttr(TC), VAttr(item_i)) \\ &selectedItem \sim P_{selection}(item_i)_{item_i \in CompatibleSet} \end{aligned} \tag{16}$$

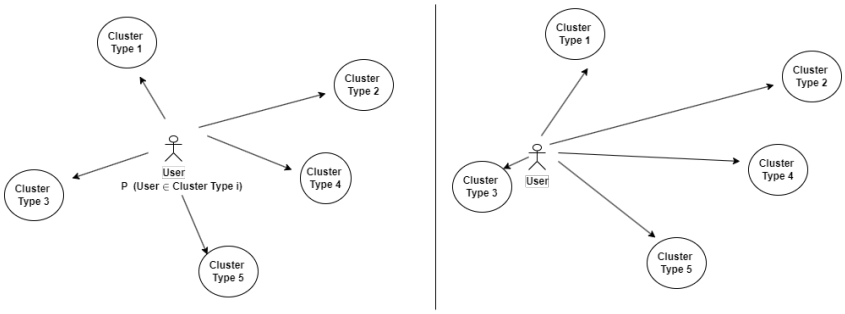

Figure 2: Showing the progress of the AI agent in classifying the user during a survey. Although the visualization is from a 2D Euclidean perspective for presentation, in reality, this is a multidimensional space. In the left part is the initial step, where the probabilities of the user being a part of each cluster are almost equal. After a sequence of questions is asked, the agent approaches one or more clusters. The probability of a user being part of a cluster is inversely proportional to the Euclidean distance to it in the hyperspace of the clusters.

```
prevClipId=-1
prevQueId=-1
needAClip=True
numQueInClip=0
GetNextAction(i, TC):
 if needAClip:
  Select the next clip C∈Clips containing features of TC
      and under constraints imposed by  prevClipId
  Show clip C
  needAClip=False
  prevClipId=C
  prevQueId=-1
  numQueInClip=0
  GetNextAction(step, targetCluster)
 else:
  if numQueInClip>MaxQueInAClip:
    needAClip=True
    break
  Select Q_i under constraints imposed by prevClipId and
      prevQueId
  if Q_i is None:
    break
  R_i = ResponseFromUser(Q_i)
  numQueInClip+=1
```

Listing 1: Pseudocode for choosing the next clip and question in the action at a given step i, and target cluster to test against, TC.

5 EVALUATION

This section describes the setup used for evaluation, some results obtained from a quantitative and qualitative evaluation, the mentioned post-survey analysis tools with some alternatives in the implementation, and finally practical observations made from prototyping and previous attempts.

5.1 Setup

The use case described in section 3 is used here for evaluation purposes. A clip describing an IB situation is shown, followed by a number of questions between 2 and 5 questions. The number of questions per survey is limited to 15-20 questions and varies depending on the user's answers and the path chosen by the AI agent evaluating the person behind the interview. These are divided equally

The database of attributes, clips, and questions is included in the repository. For the evaluation, 16 different attributes were considered, 29 clips evenly distributed among the four categories, and 35 questions, 19 of which could be asked in each of the clips.

5.2 Quantitative and quantitative evaluation

There are three research questions that we try to address in this section.
RQ1: What is the correctness of the AI agent compared with a real HR person?
To evaluate this, we took a sample of 25 people previously assessed by human HR staff and attempted to rank them after 6 months using the dataset and the AI agent (the questions and clips were new to them to avoid bias). There were a total of 435 responses to the questions. The observed comparison results follow:

1. The AI agent classified 18 out of 25, $\sim 72\%$, in the same cluster as the HR professional.
2. For 5 out of the remaining 7 persons, $\sim 20\%$, the AI agent classified in the second scored cluster the one suggested by HR. According to the formula given in Eq. (12), the measured average error difference between the first and second AI's cluster classification was ~ 0.183, suggesting that the agent was not too far.
3. The remaining two persons, $\sim 8\%$, had almost equally split probabilities to each of the four clusters.

It is however hard to tell which one was correct since even HR professionals could also have biases, and be error-prone sometimes.

RQ2: Is anchor and bias removal helpful? When using the bias and anchoring removal methods proposed in Section 4.2.2, we observed that results classified one more person in the same cluster as professionals, but on the other hand for the rest of 6 persons the average error difference increased to 0.31.
RQ3: Respondents' post-evaluation feedback After finishing the surveys, the 25 persons were asked to compare it with the ones done face-to-face. Results show that 21 out of 25 liked more this because they had more time to think, felt less pressure to give answers, and were satisfied that they had the opportunity to complete it without prior scheduling from their own comfort.

6 CONCLUSIONS

After iterating several prototype versions and applying the framework to our mentioned use case, we conclude that the proposed method of providing AI agents to conduct surveys can help organizations in two ways: a) improve the quality of the results obtained after surveys without having to invest in more staff to conduct face-to-face surveys, and b) help human professionals improve alongside AI agents by using them as assistants during a real-time survey.

ACKNOWLEDGEMENTS

This research was supported by European Union's Horizon Europe research and innovation programme under grant agreement no. 101070455, project DYNABIC.

REFERENCES

Aning, S. and Przybyła-Kasperek, M. (2022). Comparative study of twoing and entropy criterion for decision tree classification of dispersed data. *Procedia Computer Science*, 207:2434–2443. Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 26th International Conference KES2022.

Bajpai, R., Hazarika, D., Singh, K., Gorantla, S., Cambria, E., and Zimmermann, R. (2023). Aspect-sentiment embeddings for company profiling and employee opinion mining. In Gelbukh, A., editor, *Computational Linguistics and Intelligent Text Processing*, pages 142–160, Cham. Springer Nature Switzerland.

Dee, D. (2006). *Bias and data assimilation*. PhD thesis, Shinfield Park, Reading.

Iliescu, D., Ilie, A., and Ispas, D. (2011). Examining the criterion-related validity of the employee screening questionnaire: A three-sample investigation. *International Journal of Selection and Assessment*, 19(2):222–228.

Khaled, S., El-Tazi, N., and Mokhtar, H. M. O. (2018). Detecting fake accounts on social media. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 3672–3681.

Lietz, P. (2010). Research into questionnaire design: A summary of the literature. *International Journal of Market Research*, 52(2):249–272.

Mannens, E., Coppens, S., De Pessemier, T., Dacquin, H., Deursen, D., Sutter, R., and Van de Walle, R. (2013). Automatic news recommendations via aggregated profiling. *Multimedia Tools and Applications - MTA*, 63.

Ni, X., Zeng, S., Qin, R., Li, J., Yuan, Y., and Wang, F.-Y. (2017). Behavioral profiling for employees using social media: A case study based on wechat. In *2017 Chinese Automation Congress (CAC)*, pages 7725–7730.

Park, H. and Faradonbeh, M. K. S. (2022). Efficient algorithms for learning to control bandits with unobserved contexts. *IFAC-PapersOnLine*, 55(12):383–388. 14th IFAC Workshop on Adaptive and Learning Control Systems ALCOS 2022.

Rafae, A. and Erritali, M. (2023). Using a profiling system to recommend employees to carry out a project. *Electronics*, 12(16).

Samuel Farley, Daniella Mokhtar, K. N. and Niven, K. (2023). What influences the relationship between workplace bullying and employee well-being? a systematic review of moderators. *Work & Stress*, 37(3):345–372.

Schermer, B. W. (2011). The limits of privacy in automated profiling and data mining. *Computer Law and Security Review*, 27(1):45–52.

Staale Einarsen, H. H. and Notelaers, G. (2009). Measuring exposure to bullying and harassment at work: Validity, factor structure and psychometric properties of the negative acts questionnaire-revised. *Work & Stress*, 23(1):24–44.

Wibawa, A. D., Amri, A. M., Mas, A., and Iman, S. (2022). Text mining for employee candidates automatic profiling based on application documents. *EMITTER International Journal of Engineering Technology*, 10:47–62.

Xia, P., Zhang, L., and Li, F. (2015). Learning similarity with cosine similarity ensemble. *Information sciences*, 307:39–52.

Yan, T., Keusch, F., and He, L. (2018). The impact of question and scale characteristics on scale direction effects. *Survey Practice*.