Big Data, Small Personas: How Algorithms Shape the Demographic Representation of Data-Driven User Segments

Big Data, Small Personas: How Algorithms Shape the Demographic Representation of Data-Driven User Segments

JONI SALMINEN, University of Vaasa, Finland
KAMAL CHHIRANG, Fulda University of Applied Sciences, Germany
SOON-GYO JUNG, Qatar Computing Research Institute, Hamad Bin Khalifa University, Qatar
SARAVANAN THIRUMURUGANATHAN, Qatar Computing Research Institute, Hamad Bin Khalifa University, Qatar
KATHLEEN W. GUAN, Division of Psychology and Language Sciences, Faculty of Brain Sciences, University College London, United Kingdom
BERNARD J. JANSEN, Qatar Computing Research Institute, Hamad Bin Khalifa University, Qatar

Derived from the notion of algorithmic bias, it is possible that creating user segments such as personas from data results in over- or under-representing certain segments (FAIRNESS), does not properly represent the diversity of the user populations (DIVERSITY), or produces inconsistent results when hyperparameters are changed (CONSISTENCY). Collecting user data on 363M video views from a global news and media organization, we compare personas created from this data using different algorithms. Results indicate that the algorithms fall into two groups: those that generate personas with *low diversity–high fairness* and those that generate personas with *high diversity–low fairness*. The algorithms that rank high on diversity tend to rank low on fairness (Spearman's correlation: -0.83). The algorithm that best balances diversity, fairness, and consistency is Spectral Embedding. The results imply that the choice of algorithm is a crucial step in data-driven user segmentation because the algorithm fundamentally impacts the demographic attributes of the generated personas and thus influences how decision makers view the user population. The results have implications for algorithmic bias in user segmentation and creating user segments that not only consider commercial segmentation criteria but also criteria derived from ethical discussions in the computing community.

Additional Key Words and Phrases: algorithms; personas; fairness; user segmentation

1 INTRODUCTION

1.1 Conceptual Underpinnings

Personas, introduced to computer science and Human-Computer Interaction (HCI) by Cooper [19], are defined as fictitious people that represent real user and customer types. Researchers believe that personas evoke a sense of empathy [52, 54] that directs product and user experience (UX) designers, software developers, marketers, and other stakeholders to make more user-centric decisions regarding products, services, and other outputs offered to end-users and customers. Personas are, therefore, a personified form of user segmentation, i.e., dividing the overall user or customer population into demographically or behaviorally defined segments [33].

Algorithmic personas (APs) are developed from quantitative data to represent demographic and behavioral characteristics of the user base [62]. Consequently, algorithmic persona generation is the employment of algorithms and big data to create personas [46]. The promise of algorithms for persona generation was observed first by Aoyama in two articles from 2005 [7] and 2007 [8].

Authors' addresses: Joni Salminen, jonisalm@uwasa.fi, University of Vaasa, Wolffintie 34, 65200 Vaasa, Vaasa, Finland; Kamal Chhirang, kamal5chhirang@gmail.com, Fulda University of Applied Sciences, Leipziger Str. 123, 36037 Fulda, Fulda, Germany; Soon-gyo Jung, sjung@hbku.edu.qa, Qatar Computing Research Institute, Hamad Bin Khalifa University, HBKU Research Center, RC1, Doha, Qatar; Saravanan Thirumuruganathan, sthirumuruganathan@hbku.edu.qa, Qatar Computing Research Institute, Hamad Bin Khalifa University, HBKU Research Center, RC1, Doha, Qatar; Kathleen W. Guan, kathleen.guan.20@ucl.ac.uk, Division of Psychology and Language Sciences, Faculty of Brain Sciences, University College London, WC1H 0AP, London, United Kingdom; Bernard J. Jansen, bjansen@hbku.edu.qa, Qatar Computing Research Institute, Hamad Bin Khalifa University, HBKU Research Center, RC1, Doha, Qatar.

Fig. 1. Creating a persona from data using dimensionality reduction. The left-side figure shows dimensionality reduction using the UMAP algorithm on the dataset collected for this study. The right-side figure shows Siim, an algorithmic persona that corresponds to the pattern circled in the left-side figure. Different algorithms may identify different patterns in the data, but previous research has not examined how this affects the composition of the generated personas.

McGinn and Kotamraju applied an algorithmic approach for personas in their seminal article in 2008 [46]. Since then, algorithmic persona generation has become increasingly common in HCI, marketing [18], health informatics [25, 42], cybersecurity [1], video game studies [32, 64], and many other domains that use personified user segmentation for understanding users or customers.

The shift from manually created personas to APs has been characterized as transformative [50], as algorithmic persona generation can address the challenge of analyzing large collections of personified big data [71] from social media and online analytics platforms, making sense of this big user data. Manual methods are ill-equipped to analyze such amounts of user data for user segmentation and persona generation. Rapid changes in online user behavior exacerbate the challenge, as these changes require APs to be constantly updated to keep up with the user behaviors and characteristics [35]. Thus, scalable and efficient user segmentation algorithms are beneficial for transforming this data into persona profiles (see Figure 1) or other forms of user segments that describe the user populations' key behaviors and demographics.

Since the introduction of data-driven personas [46], researchers have applied a wide range of algorithms for persona generation [12, 27, 30, 32, 51, 77, 85], the most common being clustering (grouping users in a way that users in the same group are more similar to each other than to users in other groups), principal component analysis (PCA) (summarizing the user information into smaller summary indices that aim to capture most user information in a computationally efficient format), non-negative matrix factorization (NMF) (similar to PCA, NMF reduces user data from a higher dimensionality to a lower dimensionality that captures essential behavioral and/or demographic information), and latent semantic analysis (LSA) (identifying associations between a set of users and the content they are interested in by producing a set of latent concepts related to the users and content) [61]. In their review of literature, Zhu et al. [85] report the use of decision trees, exploratory factor analysis, hierarchical clustering, k-means clustering, latent semantic analysis, multidimensional scaling analysis, and weighted graphs for persona development. Minichiello et al. [51] provide a similar list of methods: cluster analysis, factor analysis, principal component analysis, and latent semantic analysis. A shared trait of these methods is the attempt to simplify the user population into segments that are then transformed ("enriched"), one way or another, into finalized persona profiles for end-users (see Figure 1).

1.2 Research Problem

This collaborative process of persona generation between humans and algorithms involves multiple challenges. While the algorithmic process is opaque to humans, they need to trust that the algorithm performs in a desirable way – that is, not exacerbating demographic biases in the data or favor one group over another when selecting traits for the personas or segments. In the following, we detail three important research problems (RPs) and our approaches to address these problems in the current study.

- First, *there is a lack of studies comparing algorithms for persona generation (**RP1**). Apart from a brief comparison by Brickey et al. [12], none of the studies address what kind of personas different algorithms produce. The lack of studies poses a major hindrance for understanding how the choice of algorithm affects persona generation and whether there is a risk for algorithmic bias in persona generation (and user segmentation in general).*
- Second, *there is a lack of using non-accuracy metrics for persona evaluation* (**RP2**). What we mean by this is that apart from [26] that evaluated inclusivity, quantitative persona studies tend to focus on evaluating personas' technical accuracy [14, 15] as opposed to what kind of personas are generated. Yet, the characteristics of the generated personas matter because persona traits (e.g., age, gender, ethnicity) risk aggravating stereotypical thinking about user populations [60, 74].
- Third, there is a general lack of *sharing resources for persona generation* (**RP3**). The lack of publicly available resources — code, data, algorithms, computational notebooks — poses a major hindrance to research on data-driven personas [61] and the broader field of user segmentation. Even though making results are resources available for others is a basic principle for scientific progress [39], few data-driven persona studies have made their resources available for others, thus limiting their contribution to the field.

To address RP1, we conduct an experimental study where we fix the (a) dataset, (b) the number of personas generated, and (c) the method of enriching the personas, only varying the algorithm that processes the baseline user data. Although several effects of algorithms on persona generation (and more broadly on customer segmentation) could be examined, including, e.g., accuracy, computational efficiency, or run-time, we focus on studying the effect of algorithm on model outputs, i.e., the generated personas. We specifically address the effect of algorithm on the demographics of the persona set – we choose demographics as the unit of analysis because of their widespread use in both persona generation and in customer segmentation overall.

To address RP2, we examine three aspects vital to decision makers interested in personas for representing their user, audience, or customer base. These aspects are *diversity* (i.e., the personas cover many unique characteristics found the user base), *fairness* (i.e., the personas truthfully reflect the underlying data on the users), and *consistency* (i.e., the algorithm retains central persona traits when changing the hyperparameters). We refer to these design goals for data-driven personas and customer segments as the **DFC criteria** (Diversity, Fairness, Consistency).

To address RP3, we make the resources (code, data, algorithms) available in an online repository [1] to further advance empirical persona generation and customer segmentation experiments. To protect business sensitive information, the data is made available in a scrambled format.

Building on above reasonings, our research question is: *How do diversity, fairness, and consistency of the generated personas' demographic attributes vary by algorithm?* The results have implications for creating more diverse, fair, and consistent personas (and other forms of user segments) from digital user data. Our results inform persona developers (and marketers) of the strengths and

[1] https://github.com/joolsa/Persona-Generation

weaknesses of different algorithms when applied to persona generation (or customer segmentation). They also provide wider implications for the potential of algorithmic bias when using algorithms for user or customer segmentation and draw attention to the use of non-commercial segmentation criteria influenced by the fairness/bias discussion going on in the computer science and HCI communities [22], which has not thus far addressed in the persona generation and customer segmentation domains. The main contributions include the following: (a) Providing the most extensive comparative study to date, comparing six different algorithms for persona generation; (b) introducing and applying novel evaluation metrics in the persona generation context, while also providing suggestions for applying different algorithms for different objectives; and (c) sharing computational resources to further persona generation research and development.

2 LITERATURE REVIEW

Table 1 shows a list of commonly applied persona generation algorithms based on a sample of 63 research papers. The sample of articles was obtained as follows. First, we searched two major academic services: Google Scholar and ACM Digital Library. We used search phrases relating to persona generation ("algorithmic personas"', "data-driven personas," "quantitative personas," "procedural personas"). The found articles were manually screened by reading the abstracts. We excluded articles if they were not (a) full research articles written in English, (b) published in a peer-reviewed journal or conference, and (c) developed personas using data-driven approaches. We also carried out snowball sampling to fully account for relevant literature [59]. In total, the database searches and snowballing yielded 163 articles for full-text screening. For the screening, we applied the same criteria (a-c). After the full-text review, a set of 63 final articles remained. We then reviewed these articles to identify the algorithm(s) applied.

Method	Description	Frequency	Examples
CA	Groups a dataset using a predetermined number of clusters. Popular approaches are partitioning based approaches such as k-means and agglomerative such as hierarchical clustering.	N=22 (34.9%)	[2, 3, 31, 48, 72]
PCA	Linear dimension-reduction algorithm used to extract information by removing non-essential elements with relatively small variation.	N=5 (7.9%)	[11, 12, 20, 73, 76],
LSA	Data analysis algorithm that uses singular value decomposition to detect hidden semantic relationships between words.	N=5 (7.9%)	[10–12, 23, 49]
NMF	Method in which data matrices are constrained as non-negative and decomposed to extract sparse and meaningful features.	N=4 (6.4%)	[4–6, 66]
LDA	A generative statistical model that models each item of a collection (typically text) as a finite mixture over an underlying set of patterns.	N=6 (9.5%)	[4, 6, 10, 21, 38, 70]

Table 1. Top 5 methods for algorithmic persona generation from literature review. Percentages are of total reviewed articles. The rest (N=21, 33.3%) are classified as 'Other'.

From the reviewed articles, we identified five commonly used algorithms for persona generation: (1) Cluster Analysis (**CA**), (2) Principal Component Analysis (**PCA**), (3) Latent Semantic Analysis (**LSA**), (4) Non-negative Matrix Factorization (**NMF**), and (5) Latent Dirichlet Allocation (**LDA**). Table 1 provides an overview of these algorithms and the studies adopting them. The use of the five

algorithms for persona generation is discussed in the following subsections. A technical description of each algorithm and their implementation for this study is given in Appendix 1. The focus of this study is not on the technical traits of these algorithms but on their implications for APs and user segmentation in general.

2.1 Clustering

Tanenbaum et al. [72] utilized **CA** (k-means) to develop personas of diabetes patients and gauge patients' readiness for adopting different medical interventions. Validation was done by calculating the Euclidean distance between the different variables and conducting Chi-squared tests. Wang et al. [76] also calculated Euclidean distances of different medical and demographic variables for their analysis of regional health data. A few articles qualitatively validated clusters by engaging subject experts as well as users themselves in reviewing the cluster results [75, 84, 85]. These individuals were tasked with assessing how representative the generated clusters were of real-life scenarios. An et al. [2, 3] applied **CA** (k-means) for persona generation, and observed that using individual-level data is expensive to collect and has concerns regarding privacy. However, the researchers did not address what kind of personas different algorithms would produce.

Kwak et al. [40], using **CA** (k-means), found the limitation that a single demographic group must fall into one persona. In contrast, various personas can be found from one demographic group, as people in the same demographic group often behave differently. A potential issue of **CA** is the "need for specialists to use expert judgment during clustering [to define hyperparameters]" (p. 19) [51]. However, this issue concerns other algorithms, too, as persona developers typically need to set the number of personas as part of the process.

Miaskiewicz et al. [49] and Mesgari et al. [48] applied hierarchical **CA** to develop clusters (and ultimately personas) of university members' experiences with learning management and institutional knowledge systems. Both studies validated their results by looking at the relations between variables within clusters. The former calculated cosine similarity (of the angles between pairs of non-zero vectors), while the latter calculated Pearson correlation (the extent of a linear relationship between two variables). Holden et al. [31] used hierarchical **CA** to analyze the medical and psycho-social variables of older adults with heart failure. Results were validated using the Kruskal-Wallis test and Welch's ANOVA to determine statistical significance between the variables as well as testing for variance, respectively. However, the researchers did not address what kind of personas different algorithms would produce.

2.2 Principal Component Analysis

PCA was used for persona generation by Sinha [69], who created personas based on users' characteristics. Wang et al. [76] used **PCA** in combination with **CA** to develop health personas of regional groups. Similarly, Brickey et al. [12] used **PCA** in combination with **LSA** and **CA** to develop personas for users of an army knowledge management system. As such, the combination of this method with others is prevalent. In fact, all of the found studies that applied **PCA** complemented it with at least one other quantitative method. As a result, validation metrics also varied and included Cohen's kappa (a statistical measure of interrater agreement of generated versus expert clusters) [12], Euclidean distances of different variables [76], Spearman's ρ (direct association between two ranked variables) [20], and a qualitative review with survey participants [73]. However, the researchers did not address what kind of personas different algorithms would produce.

2.3 Latent Semantic Analysis

Apart from the aforementioned studies [11, 12, 49], Dupree et al. [23] used **LSA** to investigate attitudes towards digital privacy and cybersecurity among university students, and recruited a

separate population (different from initial survey participants) to review the developed personas [23]. The evaluation focused on tasking individuals with self-identifying with one of the five personas and providing feedback on how realistic they were. However, the researchers did not address what kind of personas different algorithms would produce.

2.4 Non-Negative Matrix Factorization

An et al. [5, 6] used **NMF** to decompose an interaction matrix constructed from the view counts of an organization's social media content. They obtained latent content consumption patterns, associating each distinctive behavior pattern with demographics of users (i.e., age, gender, country) by a weight assessment that encodes the strength of the relationship between the demographic groups and the underlying pattern. The demographic group with the highest **NMF** weight was chosen as the representative persona demographics for its corresponding behavioral video-viewing pattern. The demographic group was then augmented with a name, occupation, photo, and other characteristics, yielding complete persona profiles. Similar approach was applied in other studies from the same research team [4, 34–36, 65, 66], with the general goal of developing personas from social media user statistics. However, the researchers did not address what kind of personas different algorithms would produce. For evaluation, rankings of the generated demographics groups in these studies were compared with true rankings based on real content engagements using Kendall rank correlation coefficient [6]. Researchers identified discriminative content for personas (i.e., content that a persona has a higher chance of engaging with compared to other personas) using a Chi-square test [3]. Also, cosine similarity has been applied to calculate among pairs of personas until the closest pairs are determined [5]. Furthermore, Salminen et al. [66] used qualitative data of social media users in a geographical region in the forms of Instagram profiles and semi-structured interviews to create what they term as "hybrid personas" (the core algorithm being NMF). Qualitative research was used to enrich further and improve their hybrid personas. However, the researchers did not address what kind of personas different algorithms would produce.

2.5 Latent Dirichlet Allocation

LDA was utilized in two studies [4, 6] to understand the viewing behavior of different demographic groups and develop user personas for a YouTube channel. In the two studies, the authors built **LDA** topic models to construct matrices (in combination with **NMF**). Dhakad et al. [21] developed buyer personas from click logs on an e-commerce portal. They employed **LDA** to model persona preferences for different occasions by sampling fashion styles and relevant fashion items across online shoppers' activity. Furthermore, Smith et al. [70] used **LDA** to develop gaming personas based on controller input data from video game analytics systems. However, the researchers did not address what kind of personas different algorithms would produce.

3 METHODOLOGY

3.1 Research Design

Conceptually, our research methodology uses different algorithms on the same dataset to produce sets of 5, 10, and 15 segments that are the bases for the personas (common numbers used in persona research [5, 6]). We complete the persona generation using a standardized approach to fully generate persona profiles with a name, picture, age, gender, country, interests, and other information. We then compare these sets of personas using three quantitative metrics (see the Evaluation Metrics section) to examine how the generated personas differ by algorithm. The process is as follows, with technical approach mentioned in parentheses:

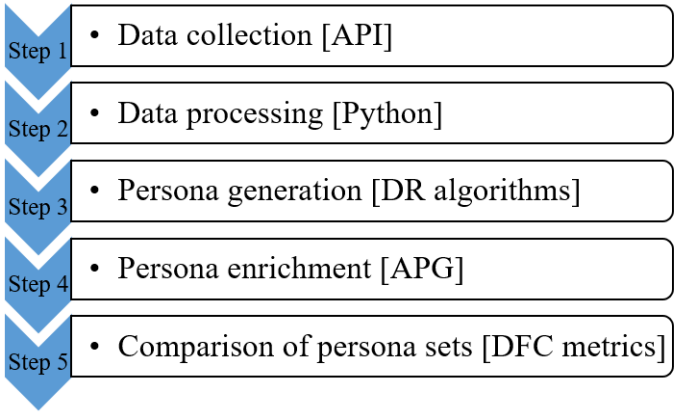


Fig. 2. The algorithmic persona generation process. **API** = Application Programming Interface. **DR = Dimensionality Reduction**. **APG** = Automatic Persona Generation. **DFC** = Diversity, Fairness, Consistency.

The experiment set-up is as follows: each algorithm processes the same data using the same number of segments ($N = \{5, 10, 15\}$), and we then enrich these segments using the same algorithmic process of assigning demographics and other information. In total, there are $3 \times 6 = 18$ persona sets (3 number levels and 6 algorithms). Using three different levels for the number of personas enables us to analyze the consistency of the results. Note that the same enrichment process is employed, producing 5, 10, and 15 personas for each algorithm. Fifteen personas is more than the conventional number of 'less than 10 personas' widely cited in the HCI literature [19, 53, 54, 57]. In this study, we consider the higher number of personas sensible, as many organizations deal with large and diverse online audiences that cannot be captured in a handful of personas. The outcome of the data collection and application of the algorithms is a collection of datasets representing the main segments or dimensions in the data.

3.2 Data Collection and Pre-Processing

We partner with an international news and media organization with 5.08M subscribers at the time of writing (April 2020) to collect a dataset that contains both behavioral (i.e., what videos were viewed and how many times) and demographic statistics about users. We collect the data from the organization's YouTube Channel by leveraging the YouTube Analytics API[2]. The dataset contains 363M views for 12.3K videos published between March 2007 and December 2019.

The justification for the dataset in terms of helping us achieve our research goals relies on the following rationale: (a) the dataset is large (typical for online user data), (b) its structure is typical for Web analytics platforms (e.g., Google Analytics and Facebook Insights provide the same output), and (c) its analysis extends beyond what can be done manually, requiring algorithmic processing to build robust personas. Also, (d) this dataset is typical for many large online businesses that generate much content or have many products/pages to offer.

The distributions of view counts by gender and age are shown in Figure 3. Geographic distribution is shown in Figure 3c.

After collecting the user data from an online analytics platform, the data is transformed into an *interaction matrix* that captures the engagement between user groups (rows) and the content items

[2]https://developers.google.com/youtube/analytics

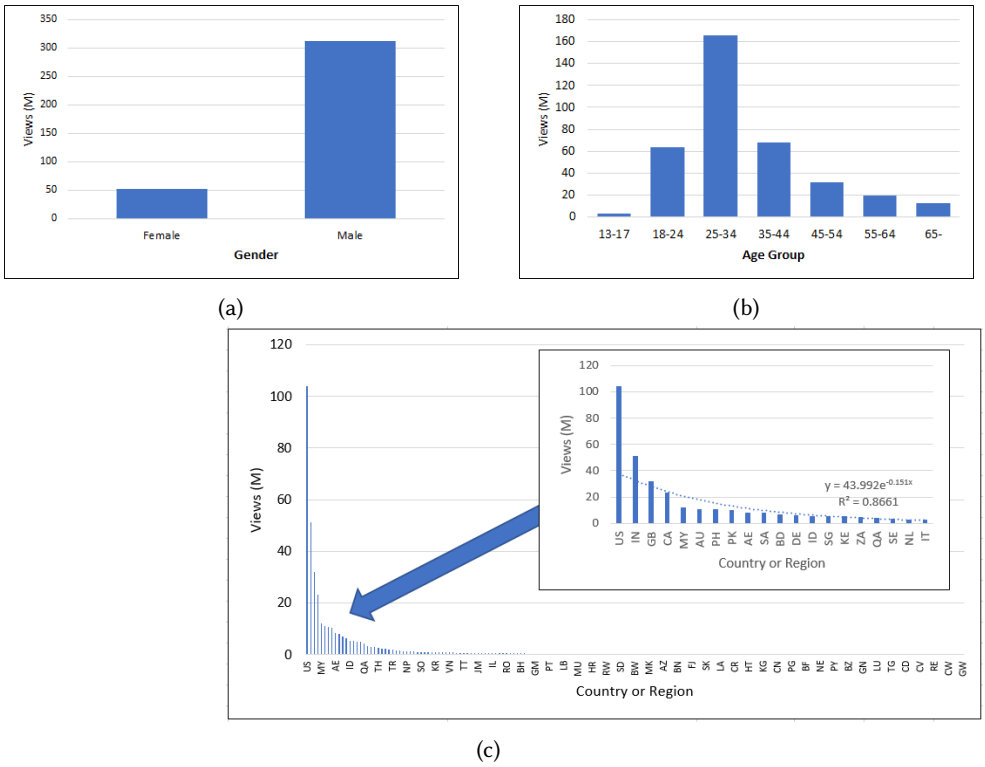


Fig. 3. Distribution of video view counts by (a) gender, (b) age, and country (c) in the baseline data. Distribution of video view counts by country and region (the smaller plot shows Top-20). The distribution indicates a highly imbalanced dataset, typical for online user data [9]. This is also observed from descriptive statistics: there are 185 countries and regions, on average, a country or region has 1.97M views but the standard deviation is 9.06M (4.6 times the average).

(columns). In this dataset, the content indicates online videos, but depending on the dataset, the content can be webpages, e-commerce products, flight destinations, or other entities of interest.

The cell values in the interaction matrix indicate the number of interactions a given demographic group (row) has for a given content (column). The demographic groups come from the online analytics and social media analytics platforms that use this grouping to aggregate user data and to protect the privacy of individual users. The age buckets used by these platforms – for example, YouTube Analytics, Google Analytics, and Facebook Insights – include 13–17, 18–24, 25–34, 35–44, 45–54, 55–64, and 65+. These groups are used to set up the interaction matrix.

For example, the demographic group "Finnish, 35-44, Male" can have 1, 200 views for "Video ABC". Thus, the values of the matrix are counts (15, 4, 000, 55, 867, ...), always either positive or zero. Many demographic groups typically have zero values for a given content, but this sparsity depends on the dataset. In our data matrix, 98.764% of the values are zeros, indicating high sparsity (the more content and demographic groups there are, the higher the sparsity tends to be because not all groups would be interested in all content). Figure 4 provides a more formal explanation using the example of NMF algorithm. In Figure 4, $\mathbf{V}$ indicates the $\mathbf{g} \times \mathbf{c}$ matrix of $\mathbf{g}$ user groups and $\mathbf{c}$ online contents. The element of $\mathbf{V}$, V_{ij}, is any number that reflects the user group G_i's interest or engagement or intent towards content C_j. In the case of Google Analytics, V_{ij} is typically a session

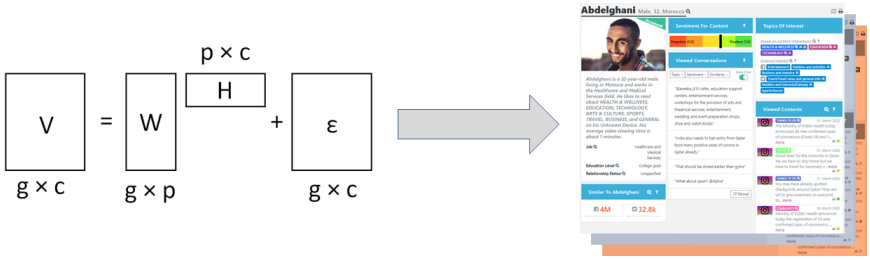


Fig. 4. Automatic persona generation. A user segmentation algorithm derives latent patterns from user data (left-hand side of the illustration). APG then enriches these latent patterns with personified information such as name, picture, demographics, and so on (right-hand side of the illustration).

for a given webpage, C_j from user group G_i. NMF decomposes $\mathbf{V}$ to two lower dimensionality matrices, $\mathbf{W}$ (demographics) and $\mathbf{H}$ (contents), which are both defined by $\mathbf{p}$ latent patterns [6]; $\mathbf{p}$ being a hyperparameter that indicates the number of personas generated. More details about NMF can be found in the seminal work by Lee and Seung [41].

Each algorithm independently processes this interaction matrix, finding p patterns (clusters, segments, components), where $p = \{5, 10, 15\}$. In this case, because the content is online videos, the segmentation aims to preserve information on the video viewing behaviors of different demographic groups. These p segments become p personas when they are *enriched* by personified information (e.g., name, picture, topics of interest) to create the final personas. Personification and enrichment are standard procedures for persona generation [46, 51, 81]; without it, the segments would remain as nameless and faceless user representations – the general benefit of personification is that human attributes increase stakeholders' empathy towards the segment the persona represents [71], while enrichment provides a more rounded, detailed information about the persona [52].

3.3 Persona Generation

For the enrichment, we use Automatic Persona Generation[3] (APG), a system for automatic persona generation. This system has numerous advantages, including standardization. What this means in our experiments is that all outputs by the algorithms undergo the same enrichment process that involves no manual intervention. For example, each piece of content is topically classified (explained in An et al. [6]), providing topics of interest for each persona.

Based on the outputs, APG chooses a representative demographic group for each latent pattern and enriches this demographic group with personified and other information (picture, name, topics of interest, quotes, etc.) to create a complete persona profile (see Figure 4). The result is a distinct set of personas based on behavioral and demographic attributes of the user population. Note that APG's procedure for assigning the demographic group is identical and deterministic – with given data and algorithm, a set of p personas will always have the same age, gender, and country when using APG for persona generation. As the only variable that changes in the process of our experiment is the algorithm, the differences in the generated personas stem from the algorithms. For a more detailed explanation and validation of APG, see An et al. [5, 6].

3.4 Choice and Implementation of Algorithms

The algorithms identified for persona generation in this study are described in Table 2. Out of the chosen algorithms, CA, PCA, NMF, and LDA have previously been applied for persona generation.

[3]link hidden for anonymous review

	Sensit. to Imbal. Data	Linearity	Comput. complexity	Interpre tability	# of hyper parameters	Topology Preservation	Impact of Sparsity	Global/ Local	Impact of Dimensionality
CA	N	L	High	High	3	N	Medium	Global	High
PCA	Y	L	Low	Low	1	Y	Low	Global	Low
NMF	Y	L	Medium	Medium	1	N	Medium	Global	Low
LDA	N	L	Medium	Low	1	N	High	Global	Medium
UMAP	N	N	Medium	High	1	Y	Low	Local	Low
SE	Y	N	High	Medium	2	Y	Low	Local	Low

Table 2. Properties of the chosen algorithms. 'L' for linear, 'N' for non-linear. Topology Preservation means whether the algorithm preserves the structure of the data: if two items are neighbors in the high dimension, will they also be neighbors in the low dimension. Global/Local: whether the algorithm tries to preserve global structure or local interactions. Usually, linear algorithms focus on global and non-linear ones on local. Impact of dimensionality: what the impact of the original dimension size is. For example, PCA and NMF work well for millions of dimensions, while CA quickly becomes inoperable.

LSA was not selected because it is a type of matrix factorization algorithm (using singular value decomposition), and we already selected NMF for testing. Because we also wanted to test new methods for persona generation, we selected two novel (as in previously not applied for APs) algorithms: Uniform Manifold Approximation and Projection (UMAP) and Spectral Embedding (SE). Given our analysis of existing algorithms, these two seem like next logical algorithms for persona generation to take on. Thus, we ended up with six algorithms to test. The chosen algorithms support the goals of this study for multiple reasons: they *(a) represent the most often used algorithms in persona research*, so comparing them is relevant; *(b) are standard approaches in computer science*, which affords implementation and replicability in future studies; *(c) involve variability (see Table 2)*, which means the comparison is likely to result in meaningful differences (e.g., in terms of linear and non-linear algorithms); and *(d) are readily available in software packages and data science libraries*, which facilitates their deployment in persona generation projects in practice. The specific implementation we use for each algorithm is explained in Appendix 1.

3.5 Evaluation Metrics

3.5.1 Reasoning. The generated persona sets are compared by three metrics, explained in the following subsections. The metrics that we propose for this problem are vital for data-driven personas and customer segmentation in general, because they address non-commercial and ethical aspects of segmentation efforts, areas that are lacking attention in these problem domains but that are broadly acknowledged as important within the computer science community [24, 28, 37].

- **Diversity** matters as a user segmentation goal because the segments should represent demographically diverse groups of people. If they do not, decision makers may end up receiving information about only select groups and thus ignore the needs and wants of other groups, posing disadvantage to those groups. As put by Drosou et al. [22], "diversity [is] an important component of a data-responsible society" (p. 73).
- **Fairness** is understood in terms of equity ("equity is defined as the quality of being fair and impartial[4]") so that the demographic segment's probability of being included in a finite number of segments shown to decision makers should reflect the segment's share of voice (i.e., representativeness, size, importance) in the baseline dataset, meaning the dataset that the user segments are created from. If a segment is highly prevalent in the baseline data but

[4]https://www.cui.edu/academicprograms/education/perfecting-the-practice/blog/post/how-to-balance-equity-equality-and-fairness

hardly visible among the segments, that would not be a fair (or equitable) representation of the data.

- **Consistency** matters because one would expect that if making changes to the algorithm's parameters, such as the number of segments created, on different runs of the algorithm the same or similar segments should be identified by the algorithm. If the algorithm instead identifies very different segments at each run, it is behaving in an unstable or random way and its outputs should be less trusted than those of a more consistent algorithm.

Regarding the interpretation of the obtained scores, high diversity indicates that a lot of different demographic groups are represented in the generated personas (segments). High fairness indicates that the generated personas (segments) correspond well with the most engaged users in the source data. High consistency indicates that the personas (segments) generated using a smaller number of personas (segments) appear also when changing the hyperparameter to higher number of personas (segments), thereby indicating higher reliability that these particular segments are important – or otherwise put, that the algorithm is not randomly selecting the segments.

3.5.2 Diversity. We use the count of unique attributes for comparing diversity. The count of unique attributes (D) is the number of persona attribute values (age groups, genders, countries) present in each persona set. For fifteen personas, a value such as $D_{15} = 16$ would indicate the algorithm designed a set of personas with 16 unique demographic attributes, e.g., 2 genders, 5 age groups, and 9 countries ($D = 2 + 5 + 9 = 16$). Note that D can be computed by considering all three demographic attributes and each demographic attribute separately. For example, $D_{age} = 5$ means the persona set contains personas from five unique age groups. D affords a straightforward interpretation of diversity between the persona sets. For example, if Persona Set A has personas from three age groups and Persona Set B has personas from six age groups, the latter is considered to be (6 - 3) / 3 = 100% more diverse (in terms of age) than the former.

3.5.3 Fairness. We use statistical parity for measuring fairness. Fairness assessments in machine learning tend to focus on prediction or classification [16, 24]. However, persona generation is not a classification task, but an unsupervised learning task. Yet, we can apply existing principles from computing studies developing tools for fairness assessment. For example, Dwork et al. [24] propose that individual fairness is defined such that similar individuals are treated similarly. In its more elementary interpretation, this implies that a member i in Group A that has the same characteristics (e.g., race, gender) than a member j in Group B will have an equal probability of succeeding (e.g., being chosen for a job). A classic example of analyzing fairness is using personal attributes such as gender or race to predict if a person is rich or not (e.g., 50k+ annual salary or not). In the case of personas or customer segmentation in general, the analogy of demographic groups being significant is the question, "what is the probability of this demographic class being selected by the algorithm among the generated personas?".

There are (at least) two ways to approach this issue [37]: equality and equity. Equality would translate to any demographic group having the same expected probability of being included in the generated persona set. An equity-based approach would translate to some groups having a higher expected probability of being included in the personas, because of their special needs or other factors. In our setting, we apply the equity-based approach, and the "other factor" is the demographic group's share of the engagements in the baseline data. This means that demographic segments with more views are expected to have a higher chance of appearing in the persona set – this is fair because it corresponds to a truthful representation of the user population. While fairness criteria is always subject to some degree of relativism, for the objective of finding the bias of algorithms, an essential question is "If 30% of the total views come from the US, then is it fair to

say that 30% of the personas in the persona set should be from the US?". If this statement is fair, then statistical parity (*SP*) is an appropriate fairness indicator [24].

Therefore, we calculate *SP* as the difference between two values:

$$SP_i = \frac{p_i}{P} - \frac{n_i}{N}, \tag{1}$$

where *SP* for a given demographic attribute i (e.g., p = "Male") is its fraction in the persona set P divided by the corresponding fraction of that demographic group's engagement counts (n) from the total N engagements. For example, if five personas are from the US in a persona set of 10, then 50% of the personas are from the US. Given that 30% of the views in the original data are from the US, the value will be $SP = 0.5 - 0.3 = 0.2$. The total *SP* is calculated by taking an average across all demographic attribute values.

3.5.4 Consistency. The notion of consistency matters because some researchers have found personas to be abstract and inconsistent [45]. For example, if the personas included in a five-persona set would be very different from the ten-persona set, this would call into question the validity of the method. To evaluate how consistent the generated personas are, we computed a Consistency Score (*CS*) for each algorithm. For this, we take the demographic groups an algorithm generates in the five persona set, and compare how many are the same as in the ten persona set; then, comparing the ten-persona set to fifteen personas. For example, if all five demographic groups are in the ten persona set, the score is 1.00. We carry out this calculation three times: comparing five personas to ten personas, ten personas to 15 personas, and five personas to 15 personas. In other words, we also calculate how many demographic groups from the ten persona set are in the 15 persona set, and how many from the five persona set are in the 15 persona set. Thus, we end up with three fractions (e.g., 1.00, 0.80, 0.20) – their average is the final *CS*. Using the number of personas is appropriate here since the number of segments is the major hyperparameter shared by all the unsupervised algorithms tested here (hyperparameter refers to an external value of a given parameter that is set by the research as opposed to being internally optimized by the algorithm itself).

Formally, the *CS* used in our study can be expressed as follows. Let us have three persona sets, denoted Set A, Set B and Set C. The number of personas in Set A is n_A. The number of personas in Set B is n_B. The number of personas in Set C is n_C. Without loss of generality, we set $n_A <= n_B <= n_C$. Comparing the three sets will result in a sub-consistency score for three combinations: Set A - Set B, Set A - Set C, and Set B - Set C. That means, we need to choose 2 in 3 sets each time for comparison of consistency. Thus, there are $\binom{3}{2}$ combinations = $\frac{3!}{2!1!}$ = 3 combinations.

Then, the *CS* for each algorithm is calculated as follows:

$$\frac{1}{3}\left(\frac{\sum_{i=1}^{n_A} \mathbb{1}_{P_{Ai}\in\{P_B\}}}{n_A} + \frac{\sum_{i=1}^{n_A} \mathbb{1}_{P_{Ai}\in\{P_C\}}}{n_A} + \frac{\sum_{i=1}^{n_B} \mathbb{1}_{P_{Bi}\in\{P_C\}}}{n_B}\right), \tag{2}$$

where P_{Ai} denotes the Persona i of Set A and $\{P_B\}$ denotes all the personas of Set B.
$\mathbb{1}_{P_{Ai}\in\{P_B\}}$ is the indicator function, which is:

$$\mathbb{1}_{P_{Ai}\in\{P_B\}} = \begin{cases} 1 \text{ if } P_{Ai} \in \{P_B\} \\ 0 \text{ if } P_{Ai} \notin \{P_B\} \end{cases} \tag{3}$$

Note that, to obtain each fraction, we divide by the lower persona set. For example, when comparing the five persona set and the ten persona set, if there are five matches (which is the maximum possible), we divide by 5 (not 10). The maximum of *CS* is, therefore, 5/5 + 5/5 + 10/10 = 3, and 3/3 = 1. Also, note that we consider a demographic group match only once. For example, if "Male, 65+, USA" appears once in 5 persona set and it appears three times in the 15 persona set, we count one match, as we consider that group represented at least once.

	CA	PCA	NMF	LDA	UMAP	SE
			5 personas			
D_{age}	4	2	3	4	3	4
D_{gender}	1	2	2	2	2	2
$D_{country}$	2	3	2	5	4	4
D_{total}	7	7	7	**11**	9	10
			10 personas			
D_{age}	5	3	3	5	6	5
D_{gender}	2	2	2	2	2	2
$D_{country}$	3	6	6	10	9	8
D_{total}	10	11	11	**17**	**17**	15
			15 personas			
D_{age}	5	3	3	6	6	6
D_{gender}	2	2	2	1	2	2
$D_{country}$	5	10	11	15	14	13
D_{total}	12	15	16	**22**	**22**	21

Table 3. Diversity results. Higher is better. For example, the value of 2 for CA, 15 personas (first column, third row from the bottom) indicates that among the 15 personas, CA generated personas from two different age groups. "Total" indicates the sum of unique age groups, genders, and countries. Highest total values of diversity bolded.

The above formula shows the special case of the *CS* metric for our study; the general case of the *CS* is provided in Appendix 2.

4 RESULTS

4.1 Diversity

The diversity results are shown in Table 3. Results from a two-factor repeated-measures ANOVA show that the algorithms significantly differ by their D values, $F(5, 10) = 12.49$, $p < .001$. The algorithms with the highest D values tend to be LDA, UMAP, and SE. A post-hoc analysis (Welch's t-test) indicates that algorithms in `Group 1`: LDA, UMAP, and SE generate significantly more unique persona attributes (M = 16) than algorithms in `Group 2`: CA, PCA, NMF (M = 11), $t(13.79) = -2.45$, $p = .028$. The observed effect size ($d = 1.16$) indicates that the magnitude of the difference between the groups is large.

For the five personas set, LDA produces personas with 57.1% more unique demographic attributes than CA, PCA, and NMF. For the ten personas set, LDA and UMAP produce personas with 70.0% more unique demographic attributes than CA and 54.5% more than PCA and NMF. For the 15 personas set, LDA and UMAP produce personas with 83.3% more unique demographic attributes than CA, 46.7% more than PCA, and 37.5% more than NMF.

In terms of age, two rare age groups are the youngest (13-17) and the oldest (65+) age group. The age group 13-17 appears in five persona sets (LDA_5, LDA_{10}, $UMAP_5$, $UMAP_{10}$, and $UMAP_{15}$), while the age group 65+ only appears in three persona sets ($UMAP_{10}$, $UMAP_{15}$, and SE_{15}). An example of a persona from this age group shown in Figure 5c. Although the age groups of these personas are less common, the countries of the generated personas tend to belong to the Top-10 countries in the baseline data, with the curious exception of Antigua and Barbuda. Interestingly, none of the persona sets contain personas from all age groups, implying that more personas beyond the number of 15 are needed to cover all age groups in the data.

(a) Conor – (SE_{15})

(b) Ville ($UMAP_{15}$)

(c) Sarah (SE_{15})

Fig. 5. "Three personas you would otherwise not see". Among the examples of personas with unique demographics, there is only one persona from Ireland **(a)**, only one from Finland **(b)**, and only one from the age group of 65+ **(c)**. The rarer demographics emerge only with certain algorithms when increasing the number of persona in a set.

All the algorithms generate personas from both genders, apart from two cases, in which none of the personas generated by the algorithm were female: CA_5, and LDA_{15} (see Table 3).

The countries of the personas show an interesting finding in that some of the algorithms generate personas that also represent fringe geographics, i.e., countries that have a very low proportion of views in the baseline data. LDA, UMAP, and SE account for most of the marginalized personas, examples shown in Figure 5. For example, view counts from users from Finland are only 0.145% of the total view counts; yet, $UMAP_{15}$ generates as Finnish persona (see Figure 5b).

4.2 Fairness

Fairness results are shown in Table 4. A two-factor repeated-measures ANOVA shows that the algorithms significantly differ by their *SP* scores, $F(5, 10) = 8.21$, $p = .003$. The lowest *SP* scores (reversely, the highest fairness) tend to be among CA, PCA, and NMF. Similarly, the persona sets significantly differ by their *SP* scores, $F(2, 10) = 19.28$, $p < .001$. This indicates the number of personas affects the diversity scores. However, unlike in the case of diversity, the best fairness values are obtained with 10 personas. As with *D*, the post-hoc analysis indicates two distinct groups emerging from the fairness results. `Group 1: CA, PCA, NMF` has significantly smaller *SP* scores ($M = 0.070$) than `Group 2: LDA, UMAP, SE` ($M = 0.118$), $t(11.84) = -3.475$, $p < .005$. The observed standardized effect size is large ($d = 1.64$). Note that smaller *SP* score indicates higher fairness (the lower the value, the closer the persona attributes are to the baseline data).

4.3 Consistency

The results of *CS* indicate that most of the tested algorithms produce consistent personas according to our definition of the persona demographic groups not changing when changing the number of personas generated. Four algorithms achieve the perfect score of $CS = 1.0$: CA, PCA, NMF, and SE. The two remaining ones behave more inconsistently, with UMAP ($CS = 0.40$) scoring higher than LDA ($CS = 0.10$). A perfect consistency implies the same personas that were part of the smaller number of personas are part of the larger number of personas as well.

The fact that all the algorithms, except LDA and UMAP, rank perfectly on *CS* implies that the algorithms behave consistently but differently – i.e., an algorithm generates the same personas in

	CA	PCA	NMF	LDA	UMAP	SE
			5 personas			
SP_{age}	**0.04**	0.11	0.05	0.19	0.12	0.14
SP_{gender}	0.14	**0.06**	**0.06**	0.26	0.26	0.26
$SP_{country}$	0.13	0.10	0.13	**0.08**	0.11	**0.08**
			10 personas			
SP_{age}	**0.03**	0.08	0.08	0.13	0.11	0.09
SP_{gender}	**0.04**	**0.04**	**0.04**	0.06	0.06	0.06
$SP_{country}$	0.10	**0.06**	**0.06**	0.10	0.07	0.07
			15 personas			
SP_{age}	**0.05**	0.09	0.10	0.10	0.07	0.09
SP_{gender}	**0.01**	0.08	0.08	0.14	0.19	0.12
$SP_{country}$	0.07	0.04	**0.02**	0.08	0.08	0.08

Table 4. Statistical parity scores. Lower is better (closer to the baseline data). The lowest numbers bolded. The values were obtained as follows: (a) First, we converted all the negative *SP* values to their absolute values. Then, (b) we calculated the mean of *SP* for each algorithm's each persona set. This was repeated (c) for all the attribute values in a demographic category. For example, Age values in the table are averages of all seven age groups, indicating how well the given algorithm represents the age group distribution in the baseline data.

	Persona 1	Persona 2	Persona 3	Persona 4	Persona 5
CA	**M 25-34 US**	**M 25-34 India**	**M 35-44 US**	M 18-24 India	**M 45-54 US**
PCA	**M 25-34 US**	**M 25-34 India**	**M 18-24 US**	**F 25-34 US**	M 25-34 Philippines
NMF	**M 25-34 US**	**M 25-34 India**	**M 35-44 US**	**M 18-24 US**	**F 25-34 US**
LDA	M 35-44 South Africa	M 13-17 India	F 13-17 USA	F 55-64 Malaysia	M 45-54 Serbia
SE	**M 45-54 US**	M 55-64 US	F 25-34 Singapore	M 55-64 Namibia	F 35-44 Grenada
UMAP	M 35-44 Brazil	M 35-44 Cote d'Ivoire	M 25-34 Venezuela	F 25-34 Venezuela	F 13-17 Dominican Republic

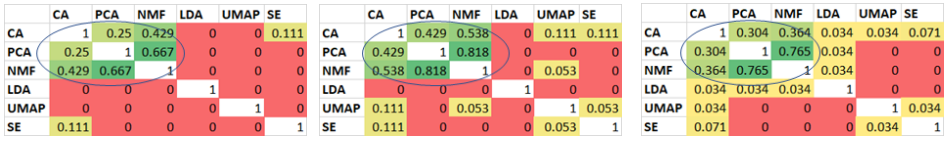
Table 5. The sets of five personas exemplifies how the personas are different, even when generating a small number. Personas appearing at least two times are bolded. US = United States.

10 and 15 persona sets than in the five persona set, but the personas differ by the algorithm. Each of the algorithms tries to identify key patterns but have different definitions of importance. Thus, the generated personas tend to be different (see Table 5).

We can further quantify *how* different the personas are by computing the Jaccard coefficient (J) for each pair of algorithms. J indicates the intersection over the union of two persona sets A and B designed by two different algorithms, which can be interpreted as similarity of persona attributes. Here, J compares the similarity of sets of personas that are defined by age-gender-country, and indicates the intersection over the union of two persona sets A and B generated by two different algorithms. J is equal to 1 if the sets are the same and 0 if they are completely different. The results in Figure 6 show that the personas outputted by different algorithms differ substantially, showing a clear clustering among CA, PCA, and NMF.

The inconsistency of the UMAP and LDA can partially explain the high D scores of these algorithms. In other words, the inconsistency of LDA and UMAP is because they choose novel demographic groups when generating the persona segments. This proposition is supported by the observed strong negative relationship between CS and D (ρ = -0.823), which implies diversity-consistency trade-off.

This trade-off is defined as follows: If optimizing for diversity, fairness and consistency decrease. Conversely, if optimizing for fairness or consistency, there will be less diversity in the personas. This



	CA	PCA	NMF	LDA	UMAP	SE
CA	1	0.25	0.429	0	0	0.111
PCA	0.25	1	0.667	0	0	0
NMF	0.429	0.667	1	0	0	0
LDA	0	0	0	1	0	0
UMAP	0	0	0	0	1	0
SE	0.111	0	0	0	0	1

(a) 5 personas

	CA	PCA	NMF	LDA	UMAP	SE
CA	1	0.429	0.538	0	0.111	0.111
PCA	0.429	1	0.818	0	0	0
NMF	0.538	0.818	1	0	0.053	0
LDA	0	0	0	1	0	0
UMAP	0.111	0	0.053	0	1	0.053
SE	0.111	0	0	0	0.053	1

(b) 10 personas

	CA	PCA	NMF	LDA	UMAP	SE
CA	1	0.304	0.364	0.034	0.034	0.071
PCA	0.304	1	0.765	0.034	0	0
NMF	0.364	0.765	1	0.034	0	0
LDA	0.034	0.034	0.034	1	0	0
UMAP	0.034	0	0	0	1	0.034
SE	0.071	0	0	0	0.034	1

(c) 15 personas

Fig. 6. Pairwise Jaccard coefficient values for personas generated from the data. The values indicate the overlap of the personas in terms of age, gender, and country. The circles illustrate the tendency of CA, PCA, and NMF to generate similar personas across the different generations. In contrast, the behavior of LDA, UMAP, and SE is more sporadic.

trade-off implies that diversity and fairness/consistency are conflicting design goals for algorithmic persona generation, at least when increasing the number of personas. We can tackle this trade-off by taking the average rank of the algorithms and the DFC metrics to assign a composite rank score for each algorithm. When doing so, SE appears as the most "balanced" algorithm (see Table 6), followed by NMF. CA, while being the most commonly used algorithm in DDPD research, ranks the lowest on this composite comparison.

Depending on the use case, persona developers may want to prioritize certain design goals, such as diversity over fairness (or vice versa). For this, the computations can be further developed by introducing three parameters: Diversity Penalty (α), Fairness Penalty (β), and Consistency Penalty (γ). Let us say that for a particular use case, the persona developers want to increase fairness, but at the same time, they want to main a high diversity and consistency of the personas. Thus, they consider all metrics important, but fairness three times as important as the two others. Accordingly, they set α and γ to 0.20, and β to 0.60, such that the

$$MRS_i = R_{D,i} \times 0.20 + R_{SP,i} \times 0.60 + R_{CS,i} \times 0.20, \tag{4}$$

where the Mean Rank Score MRS of algorithm i is calculated as a weighted sum of Rank R of i for each metric.

Using these penalty parameters to compute the MRS, the ranking of the algorithms is now in favor of NMF, with SE falling a shared second position with PCA (see Table 6). Note that these parameters are presented as examples only; future work should conduct a proper sensitivity analysis. Nonetheless, considering the DFC design goals as "weights" for the algorithms is intuitive.

Algorithm	**Rank (MRS) unweighted**	Algorithm	**Rank (MRS) weighted**
SE	1 (2.50)	NMF	1 (0.83)
NMF	2 (2.83)	PCA	2.5 (0.90)
PCA	3 (3.17)	SE	2.5 (0.90)
LDA	4 (4.08)	CA	4 (1.37)
CA	5 (4.17)	LDA	5 (1.48)
UMAP	6 (4.25)	UMAP	6 (1.52)

Table 6. Unweighted (left-hand side) and weighted (right-hand side) rankings of the algorithms based on average rank by DFC criteria. Blue and red colors indicate changing of the "best" algorithm when applying weights. Weights provide a simple technique for persona creators to adjust the DFC criteria according to their design goals.

5 DISCUSSION

5.1 Research Contribution

As far as we know, this is the most extensive study to experiment with different algorithms for persona generation to date. Overall, our results suggest that the tested algorithms can be categorized into two groups: (a) those with *low diversity and high fairness* (CA, PCA, NMF), and (b) those with *high diversity and low fairness* (LDA, UMAP, and SE). This relationship is supported by the strong negative correlation (Spearman's ρ = -0.83) between the diversity and fairness rankings of the algorithms. Furthermore, the results indicate that the highest diversity is consistently achieved with 15 personas rather than with 10 or 5 personas. The same cannot be said for fairness; even though the number of personas has a statistically significant effect on *SP*, the best average performance is obtained with ten personas. These findings have several implications. First, concerning algorithmic persona generation, our findings expand the algorithmic persona work by Chapman et al. [14, 15] and Brickey et al. [11, 12]. In regards to the former, we provide quantitative metrics for persona evaluation [15] and evaluate demographic variables rather than coverage or prevalence [14]. Our study addresses the lack of standardized metrics for persona evaluation [61] by using three relevant metrics to assess persona generation outcomes. We also confirm the findings that different algorithms tend to "disagree" [12] – that is, design different personas from the same baseline data. While Brickey et al. [12] tested two algorithms (CA, PCA), our study considers these two and four additional algorithms. Also, Brickey et al. [12] used inter-rater agreement (Cohen's κ) for evaluation, while we use metrics specifically tailored for persona generation goals.

In terms of findings, it is interesting that the three methods that maximize diversity – LDA, UMAP, and SE – are also the three methods most rarely applied in the persona literature (see Table 1). No previous study uses SE and UMAP to generate APs. LDA has been used previously [5, 6], but CA, PCA, and NMF are dominant methods in persona generation. Our findings imply that the research and practice of persona generation (and user segmentation) benefits from experimentation with novel algorithms, as these novel approaches can result in fairer or more diverse persona sets.

Technically oriented researchers often see the proclaimed objectivity of algorithms as an advantage relative to manual persona generation and user segmentation [46, 50, 62, 84]. However, our findings suggest that it is possible that algorithmically created segments might not be any more diverse, fair, or consistent than those created completely by humans using manual means. Instead, algorithmic personas can also be subject to demographic biases that may originate from multiple sources, such as data distributions, the way algorithms process the data (in the mathematical sense), or from the assigned hyperparameters [28].

To this end, the use of the DFC metrics shifts persona evaluation away from the traditional *technical metrics* (e.g., perplexity, accuracy, loss, error, etc.) towards evaluating the outputs of persona generation in terms of *the kind of personas* the algorithm design. The metrics we use support the design goals of diverse, fair, and consistent personas, taking a step towards ethically robust APs and user segments that portray the diversity of the user base in an accurate manner [60]. The connection of persona generation and user segmentation to algorithmic fairness is an important contribution that should be further expanded upon in computational studies dealing with customer segmentation using big data. Hence, the results suggest there is a need for discussion around algorithmic bias in customer segmentation literature. As we suggest, these concerns can be addressed and awareness to them created by leveraging new metrics inspired by the on-going fairness/bias discussion in the computing community.

Concerning the real-world impact of our findings, there are crucial observations to be made about the large impact that the choice of one algorithm over another has on the composition of the generated personas or user segments. Given that customer segmentation permeates almost

every organization on the planet, there is a crucial need for awareness on how a simple change of algorithm can drastically alter the outcomes obtained from the same customer data. Because firms and decision makers are not looking to offer "everything for everyone" but instead tailor their offerings based on segments, whether using personified segments (i.e., personas) or some other types of segments, the impact of the chosen algorithm seems to be not trivial but drastic.

This observation puts pressure on organizations from two sides: first, (a) which persona generation or customer segmentation algorithm should they choose for a given situation? Second, (b) given the obscure behavior of algorithms for persona generation and customer segmentation, is their use dangerous and potentially misleading? Should new segmentation techniques be developed from scratch? Because of these fundamental questions, we expect this study not to be an isolated incidence, but part of a larger research agenda on improving algorithm persona generation and customer segmentation. While fairness frameworks and acronyms like FAT (**F**airness, **A**ccountability, and **T**ransparency), FAIR (**F**indable, **A**ccessible, **I**nteroperable, and **R**eusable) [78], FATES (**F**airness, **A**ccountability, **T**ransparency, **E**thics, **S**afety and Security) [79], and EQUATE (**E**quity, **A**ccountability, **T**rust, and **E**xplainability) [55] have been developed by the research community and industry actors to scrutinize the use of algorithmic decision-making in many fields, the application of these frameworks or concepts in the domain of APs (or user segmentation in general) is lacking. Hence, our study makes an important contribution to investigating fairness in the context of algorithmic user segmentation. Focusing on *what type of personas* are created is essential because algorithmic persona studies often assume that the use of algorithms and quantitative data prevents persona developers from injecting their biased interpretations into the created personas [50]. However, if the use of algorithms would involve aspects of unfair, inaccurate, or inconsistent personas, this would present a major issue for the ethics of persona generation or other type of algorithmic user segmentation.

5.2 Algorithms as Conveyors of Partial Truths About the User Base

Our results imply that the choice of an algorithm has a fundamental impact on the personas generated. This is an important discovery since APs tend to have an air of objectivity, credibility, and truthfulness in the eyes of stakeholders [68]. Our findings imply that attributing these properties to data-driven user segmentation might not be justified at all times. More precisely, it seems impossible to argue that any of the applied algorithms captures the "truth" about the users. Instead, each algorithm focuses on certain facets of the user population.

Moreover, the complexity of the algorithms (from a mathematical point of view) typically makes it intractable to understand why a specific trait was chosen over another. This intractability concerns personas and all user segmentation efforts carried out using algorithms. The use of algorithms is always "biased" in the sense that different algorithms produce different outputs. Nevertheless, the use of algorithms is always "objective" because, given the same data and the same parameters, an algorithm always produces the same set of personas. Therefore, it is crucial to disentangle the concepts of *truth* and *objectivity* – they refer not to the same thing.

Despite this, researchers *can* define *design goals* and desiderata for algorithmic personas. Perhaps even more so because there is no one perfect method for persona generation. In the absence of this perfect method, the focus should be on what kind of personas are being designed by different algorithms: are they diverse, are they fair, are they consistent?

The outcome of using the same data, but getting different results is a conundrum for the application of data science methods for persona generation and user segmentation in general. It stresses not only the "design power" that the algorithms have but involves a more fundamental, perhaps unanswerable question of which algorithm correctly portrays the users. This question, associated with epistemological standpoints such as truthfulness and objectivity of algorithms when creating

data-driven user segments, can be traced back to the discussion on the (im)possibility of scientific verifiability and falsifiability of personas and data-driven user segments [15].

Within the scope of this study, we are unable to provide definite answers in this regard. However, we express the concern that a precarious use of quantitative methods, coupled with stakeholders' overconfidence in algorithmic superiority due to the mystique involved with quantitative data and mathematical formulas [68], can result in a disservice. In turn, broader awareness of there not necessarily being "one truth" about the user segments is likely to increase confusion among end-users of personas who the "real" personas are. Perhaps what is needed is to switch the argument for algorithmic persona generation from one single objectivity to relative subjectivity: *here is what an algorithm has to say about your users – but it is not the whole truth.*

5.3 Practical Implications

If all algorithms generate different personas, which one should a decision maker choose? An answer arises, on one hand, from preference and context for which the personas are developed (i.e., *the purpose*) and, on the other hand, from the intimate understanding of the nature of different algorithms when exposed to specific data (i.e., *the know-how*). Persona developers may choose to maximize diversity, fairness, or consistency to generate persona sets that are most applicable for their use case. Our results show that using different target metrics yields mixed results. The choice of the algorithm depends on the goal of persona generation. In particular,

- To optimize for **diversity**, use **LDA**, **UMAP**, or **SE**.
- To optimize for **fairness**, use **CA** or **NMF**.
- To optimize for **consistency**, use **CA**, **PCA**, **NMF**, or **SE**.
- When accounting for all the three criteria, use **SE** or **NMF**.

Our results show that different algorithms design different personas from the same user data. Thus, the practitioner's choice of algorithm ultimately results in different personas. This implies that the choice of the algorithm should not be taken lightly. More precisely, the practitioner faces two important choices: (a) the choice of the algorithm for persona generation, and (b) the choice of the hyperparameters for the selected algorithm. The ethical implication is that rather than hiding these choices under the parlance of "statistical", "objective", and "data", transparency and discussion of the pros and cons of these choices should be undertaken by the wielders of the algorithms. If one cannot explain it, one probably should not be using it. Moreover, an important guideline is to consider the *goal* of the customer segmentation or persona generation exercise in the first place – for example, if one seeks to get as varied understanding of the user population as possible, then using an algorithm that maximizes diversity would be beneficial. If, instead, one seeks to get a tight understanding of the most engaged segments, a fairness-based algorithm would be applicable. The paradigm of "here is our data – algorithm, please show our segments" needs to be revised to "here is our data AND our goal – algorithm, please show our segments."

5.4 Implications for Segmentation Researchers

Concerning replicability and applicability, the fact that the dominant online platforms tend to output a data structure that is compatible with persona generation means that, by using this data and publicly available data science algorithms, anyone with access to data and necessary programming skills can generate personas. Thus, the practical implications of this study range across many industries and contexts, like the method of personas itself. For example, from YouTube Analytics, one can collect videos and their view counts; from Google Analytics, pages and their session counts; from Facebook Ads, the ads and their interaction metrics (views, clicks, purchases).

To advance the use of data-driven personas, we share our source code and data (with content IDs masked and view counts randomized for protecting business-sensitive information). Researchers and practitioners can obtain these resources via the code repository[5]. Sharing data, algorithms, and code is crucial for achieving progress within the user segmentation research and practice [61] and we hope that research contributes to setting an example of making computational materials and resources for persona generation and customer segmentation available to both research and practitioner communities.

5.5 Limitations and Future Research Directions

There are several directions to pursue from our findings.

First, the properties of the algorithms most likely explain some of the results. For example, some algorithms may be more sensitive to imbalanced data. While Table 2 provides descriptive information of the tested algorithms, we did not test how these properties of the algorithms affect the results. This is primarily because of the parsimonious experimental setting that focused on observing the effects of using different algorithms on the demographic composition of the generated personas. More work is required to establish explanations as to why the algorithms behave differently, but providing such explanations is beyond the scope of this work and thus left for future research.

Second, future research should also be directed towards a systematic understanding of how the dataset properties affect the results. These properties may include (a) *prevalence of different user demographics* (i.e., the number of rows), (b) *distributions of engagement across those user attributes*, (c) *size of content* (e.g., small organizations vs. big content producers), and (d) *sparsity of the interaction matrix*. The more datasets one would analyze, given they contain variation along with these properties, the better one would understand the relationship of data properties and the personas generated by the algorithms.

Third, another interesting question is if personas generated by different algorithms could be more/less similar under a different parametrization. In this study, we kept the hyperparameters (mainly the number of personas generated) fixed to control the effect of number on the results, but future work could investigate how the manipulation of the algorithms' hyperparameters affects persona generation. We limited the persona sets to three since the main focus of the study was on the algorithms and not the number of personas. We chose the numbers of personas for the sets (five, ten, fifteen) based on the fact that previous research tends to favor a relatively low number of personas in a set. Nevertheless, both of these parameters could be altered by (a) comparing more sets, and (b) increasing the number of personas beyond fifteen. Such extended analyses would help better understand the effect of the number hyperparameter on the algorithmic personas.

Fourth, there is a grave need for explainability and interpretation of unsupervised algorithms such as the ones we deployed in our study – generally, this a challenge for the whole ML domain (see, e.g., [29]). As our findings show, due to the unpredictable nature of most algorithms, explaining their "thought process" of choosing the specific set of demographic segments should be scrutinized in dedicated studies. Related to explainability and to the "disagreement" among the algorithms about what segments to highlight, there is a lingering question about the design of entirely new data-driven persona and customer segmentation algorithms. Here, using interactive and intelligent system functionalities alongside with computational techniques such as top-N picking and outlier detection can, we believe, yield results that are simpler and provide more meaningful and interpretable results for stakeholders than the currently used black-box algorithms.

[5]https://github.com/joolsa/Persona-Generation

Fifth, other algorithms beyond the ones we tested could be experimented with. We chose the specific algorithms based on their commonness in persona generation and customer segmentation, but more advanced or differently designed algorithms in the current body of computer science could deliver complementing results. The chosen algorithms also include derivative version such as constrained NMF [13] that could be explored in future studies.

Finally, evaluation of personas and customer segments is generally considered an on-going research area with room for contribution [15, 63]. While we propose metrics to quantify "good personas" according to certain design goals, more quantitative metrics for persona evaluation could be devised, which remains an important goal for future research. User segmentation research could also investigate ways to incorporate the metrics directly into the algorithm's objective functions, rather than focusing on a post-hoc analysis of the personas. New algorithms could make it possible for creators to specify their DFC targets prior to persona or segment generation.

6 CONCLUSION

Persona generation via algorithms is widely considered as objective in contrast to manual persona generation, but it is largely overlooked that different algorithms actually generate very different personas. Our results indicate two groups of algorithms that produce very different outcomes for persona generation: algorithms that generate personas with low diversity/high fairness and those that generate personas with high diversity/low fairness. Most algorithms produce consistent results independent of the number of personas. Persona developers should take care when selecting an algorithm for persona generation (or user segmentation in general), as the algorithm's choice impacts the diversity, fairness, and consistency of the personas. The fact that the algorithms create different personas from the same user data implies that algorithms have more influence in the user segmentation process than commonly understood.

REFERENCES

[1] Ala Sarah Alaqra and Erik Wästlund. 2019. Reciprocities or Incentives? Understanding Privacy Intrusion Perspectives and Sharing Behaviors. In *HCI for Cybersecurity, Privacy and Trust*, Abbas Moallem (Ed.). Vol. 11594. Springer International Publishing, Cham, 355–370. https://doi.org/10.1007/978-3-030-22351-9_24 Series Title: Lecture Notes in Computer Science.

[2] Jisun An, Haewoon Kwak, and B. J. Jansen. 2016. Towards Automatic Persona Generation Using Social Media. In *Proceedings of Third International Symposium on Social Networks Analysis, Management and Security (SNAMS 2016), The 4th International Conference on Future Internet of Things and Cloud*. IEEE, Vienna, Austria.

[3] Jisun An, Haewoon Kwak, and B. J. Jansen. 2016. Validating Social Media Data for Automatic Persona Generation. In *Proceedings of Second International Workshop on Online Social Networks Technologies (OSNT-2016), 13th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA)*. IEEE, Agadir, Morocco.

[4] Jisun An, Haewoon Kwak, and Jim Jansen. 2017. Personas for Content Creators via Decomposed Aggregate Audience Statistics. 632–635. https://doi.org/10.1145/3110025.3110072

[5] Jisun An, Haewoon Kwak, Soon-gyo Jung, Joni Salminen, and Bernard J. Jansen. 2018. Customer segmentation using online platforms: isolating behavioral and demographic segments for persona creation via aggregated user data. *Social Network Analysis and Mining* 8, 1 (2018). https://doi.org/10.1007/s13278-018-0531-0

[6] Jisun An, Haewoon Kwak, Joni Salminen, Soon-gyo Jung, and Bernard J. Jansen. 2018. Imaginary People Representing Real Numbers: Generating Personas from Online Social Media Data. *ACM Transactions on the Web (TWEB)* 12, 3 (2018).

[7] M. Aoyama. 2005. Persona-and-scenario based requirements engineering for software embedded in digital consumer products. In *Proceedings of the 13th IEEE International Conference on Requirements Engineering (RE'05)*. Washington, DC, USA, 85–94. https://doi.org/10.1109/RE.2005.50

[8] M. Aoyama. 2007. Persona-Scenario-Goal Methodology for User-Centered Requirements Engineering. In *Proceedings of the 15th IEEE International Requirements Engineering Conference (RE 2007)*. Delhi, India, 185–194. https://doi.org/10.1109/RE.2007.50

[9] Zlatka Avramova, Sabine Wittevrongel, Herwig Bruneel, and Danny De Vleeschauwer. 2009. Analysis and modeling of video popularity evolution in various online video content systems: Power-law versus exponential decay. In *2009*

First International Conference on Evolving Internet. IEEE, 95–100.

[10] David Bamman, Brendan O'Connor, and Noah A Smith. 2013. Learning Latent Personas of Film Characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*. Sofia, Bulgaria, 10.

[11] Jon Brickey, Steven Walczak, and Tony Burgess. 2010. A Comparative Analysis of Persona Clustering Methods. In *AMCIS 2010 Proceedings (Paper 217)*.

[12] J. Brickey, S. Walczak, and T. Burgess. 2012. Comparing Semi-Automated Clustering Methods for Persona Development. *IEEE Transactions on Software Engineering* 38, 3 (May 2012), 537–546. https://doi.org/10.1109/TSE.2011.60

[13] Hao Cai, Bo Liu, Yanshan Xiao, and LuYue Lin. 2020. Semi-supervised multi-view clustering based on orthonormality-constrained nonnegative matrix factorization. 536 (2020), 171–184. Publisher: Elsevier.

[14] Christopher N. Chapman, E. Love, R.P. Milham, P. ElRif, and J.L. Alford. 2008. Quantitative evaluation of personas as information. In *Human Factors and Ergonomics Society 52nd Annual Meeting* (2008). 1107–1111.

[15] Christopher N. Chapman and Russell P. Milham. 2006. The Personas' New Clothes: Methodological and Practical Arguments against a Popular Method. In *Human Factors and Ergonomics Society Annual Meeting* (2006), Vol. 50. 634–636.

[16] Alexandra Chouldechova. 2017. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data* 5, 2 (2017), 153–163. Publisher: Mary Ann Liebert, Inc. 140 Huguenot Street, 3rd Floor New Rochelle, NY 10801 USA.

[17] Andrzej Cichocki, Rafal Zdunek, Anh Huy Phan, and Shun-ichi Amari. 2009. *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*. John Wiley & Sons. Google-Books-ID: KaxssMiWgswC.

[18] Michael F. Clarke. 2015. The Work of Mad Men that Makes the Methods of Math Men Work: Practically Occasioned Segment Design. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, Seoul, Republic of Korea, 3275–3284.

[19] Alan Cooper. 2004. *The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity (2nd Edition)*. Pearson Higher Education.

[20] Duy Dang-Pham, Siddhi Pittayachawan, and Mathews Nkhoma. 2015. Demystifying online personas of Vietnamese young adults on Facebook: A Q-methodology approach. *Australasian Journal of Information Systems* 19, 0 (Nov. 2015). https://doi.org/10.3127/ajis.v19i0.1204

[21] Lucky Dhakad, Mrinal Das, Chiranjib Bhattacharyya, Samik Datta, Mihir Kale, and Vivek Mehta. 2017. SOPER: Discovering the Influence of Fashion and the Many Faces of User from Session Logs using Stick Breaking Process. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management - CIKM '17*. ACM Press, Singapore, Singapore, 1609–1618. https://doi.org/10.1145/3132847.3133007

[22] Marina Drosou, H.v. Jagadish, Evaggelia Pitoura, and Julia Stoyanovich. 2017. Diversity in Big Data: A Review. 5, 2 (2017), 73–84. https://doi.org/10.1089/big.2016.0054 Publisher: Mary Ann Liebert, Inc., publishers.

[23] Janna Lynn Dupree, Richard Devries, Daniel M. Berry, and Edward Lank. 2016. Privacy Personas: Clustering Users via Attitudes and Behaviors Toward Security Practices. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 5228–5239. https://doi.org/10.1145/2858036.2858214 event-place: San Jose, California, USA.

[24] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*. 214–226.

[25] A. Gonzalez De Heredia, J. Goodman-Deane, S. Waller, P. J. Clarkson, D. Justel, I. Iriarte, and J. Hernández. 2018. Personas for policy-making and healthcare design. In *Proceedings of International Design Conference, DESIGN*, Vol. 6. 2645–2656. http://publications.eng.cam.ac.uk/1037918/

[26] Joy Goodman-Deane, Sam Waller, Dana Demin, Arantxa González-de Heredia, Mike Bradley, and John P. Clarkson. 2018. Evaluating Inclusivity using Quantitative Personas. https://doi.org/10.21606/drs.2018.400

[27] Hang Guo and Khasfariyati Binte Razikin. 2015. Anthropological User Research: A Data-Driven Approach to Personas Development. In *Proceedings of the Annual Meeting of the Australian Special Interest Group for Computer Human Interaction (OzCHI '15)*. ACM, New York, NY, USA, 417–421. https://doi.org/10.1145/2838739.2838816 event-place: Parkville, VIC, Australia.

[28] Sara Hajian, Francesco Bonchi, and Carlos Castillo. 2016. Algorithmic bias: From discrimination discovery to fairness-aware data mining. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (2016). ACM, 2125–2126.

[29] Sona Hasani, Saravanan Thirumuruganathan, Nick Koudas, and Gautam Das. 2021. Shahin: Faster Algorithms for Generating Explanations for Multiple Predictions. In *Proceedings of the 2021 International Conference on Management of Data* (New York, NY, USA, 2021-06-09) *(SIGMOD/PODS '21)*. Association for Computing Machinery, 2235–2243. https://doi.org/10.1145/3448016.3457332

[30] Ilyena Hirskyj-Douglas, Janet C Read, and Matthew Horton. 2017. Animal Personas: Representing Dog Stakeholders in Interaction Design. In *Proceedings of the 31st British Computer Society Human Computer Interaction Conference (HCI '17)*. BCS Learning & Development Ltd., Swindon, UK, 37:1–37:13. https://doi.org/10.14236/ewic/HCI2017.37 event-place: Sunderland, UK.

[31] Richard J. Holden, Anand Kulanthaivel, Saptarshi Purkayastha, Kathryn M. Goggins, and Sunil Kripalani. 2017. Know thy eHealth user: Development of biopsychosocial personas from a study of older adults with heart failure. *International Journal of Medical Informatics* 108 (Dec. 2017), 158–167. https://doi.org/10.1016/j.ijmedinf.2017.10.006

[32] C. Holmgard, M. C. Green, A. Liapis, and J. Togelius. 2018. Automated Playtesting with Procedural Personas with Evolved Heuristics. *IEEE Transactions on Games* PP, 99 (2018), 1–1. https://doi.org/10.1109/TG.2018.2808198

[33] Angus Jenkinson. 1994. Beyond segmentation. 3, 1 (1994), 60–72.

[34] Soon-gyo Jung, Joni Salminen, Jisun An, Haewoon Kwak, and B. J. Jansen. 2018. Automatically Conceptualizing Social Media Analytics Data via Personas. San Francisco, California, USA.

[35] Soon-gyo Jung, Joni Salminen, and Bernard J. Jansen. 2019. Personas Changing Over Time: Analyzing Variations of Data-Driven Personas During a Two-Year Period. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems (CHI EA '19)*. ACM, Glasgow, UK, LBW2714:1–LBW2714:6. https://doi.org/10.1145/3290607.3312955 event-place: Glasgow, Scotland Uk.

[36] Soon-gyo Jung, Joni Salminen, Haewoon Kwak, Jisun An, and Bernard J. Jansen. 2018. Automatic Persona Generation (APG): A Rationale and Demonstration. In *Proceedings of the 2018 Conference on Human Information Interaction & Retrieval*. ACM, New Brunswick, NJ, USA, 321–324. https://doi.org/10.1145/3176349.3176893

[37] Jon Kleinberg, Jens Ludwig, Sendhil Mullainathan, and Ashesh Rambachan. 2018. Algorithmic fairness. In *Aea papers and proceedings* (2018), Vol. 108. 22–27. https://doi.org/DOI:10.1257/pandp.20181018

[38] Sean Kross and Philip J. Guo. 2018. Students, systems, and interactions: synthesizing the first four years of learning@scale and charting the future. In *Proceedings of the Fifth Annual ACM Conference on Learning at Scale (L@S '18)*. Association for Computing Machinery, London, United Kingdom, 1–10. https://doi.org/10.1145/3231644.3231662

[39] Thomas S. Kuhn. 1970. *The structure of scientific revolutions*. University of Chicago press.

[40] Haewoon Kwak, Jisun An, and B. J. Jansen. 2017. Automatic Generation of Personas Using YouTube Social Media Data. In *Proceedins of the Hawaii International Conference on System Sciences (HICSS-50)*. Waikoloa, Hawaii, USA, 833–842.

[41] Daniel D. Lee and Sebastian H. Seung. 1999. Learning the Parts of Objects by Non-negative Matrix Factorization. *Nature* 401, 6755 (1999), 788–791.

[42] Cynthia LeRouge, Jiao Ma, Sweta Sneha, and Kristin Tolle. 2013. User profiles and personas in the design and development of consumer health technologies. *International journal of medical informatics* 82, 11 (2013), e251–e268.

[43] Bin Luo, Richard C. Wilson, and Edwin R. Hancock. 2003. Spectral embedding of graphs. *Pattern Recognition* 36, 10 (Oct. 2003), 2213–2230. https://doi.org/10.1016/S0031-3203(03)00084-0

[44] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research* 9, Nov (2008), 2579–2605. http://www.jmlr.org/papers/v9/vandermaaten08a.html

[45] Tara Matthews, Tejinder Judge, and Steve Whittaker. 2012. How Do Designers and User Experience Professionals Actually Perceive and Use Personas?. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 1219–1228. https://doi.org/10.1145/2207676.2208573

[46] Jennifer Jen McGinn and Nalini Kotamraju. 2008. Data-driven persona development. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, Florence, Italy, 1521–1524. https://doi.org/10.1145/1357054.1357292

[47] Leland McInnes, John Healy, and James Melville. 2018. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv:1802.03426 [cs, stat]* (Dec. 2018). http://arxiv.org/abs/1802.03426 arXiv: 1802.03426.

[48] Mostafa Mesgari, Chitu Okoli, and Ana Ortiz de Guinea. 2015. Affordance-based User Personas : A mixed-method Approach to Persona Development. In *AMCIS 2015 Proceedings*. https://aisel.aisnet.org/amcis2015/HCI/GeneralPresentations/1

[49] Tomasz Miaskiewicz, Tamara Sumner, and Kenneth A. Kozar. 2008. A latent semantic analysis methodology for the identification and creation of personas. In *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08*. ACM Press, Florence, Italy, 1501. https://doi.org/10.1145/1357054.1357290

[50] T. Mijač, M. Jadrić, and M. Ćukušić. 2018. The potential and issues in data-driven development of web personas. In *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (2018). 1237–1242. https://doi.org/10.23919/MIPRO.2018.8400224

[51] Angela Minichiello, Joel R. Hood, and Derrick Shawn Harkness. 2018. Bringing User Experience Design to Bear on STEM Education: A Narrative Literature Review. *Journal for STEM Education Research* 1, 1-2 (2018), 7–33.

[52] Lene Nielsen. 2019. *Personas - User Focused Design* (2nd ed. 2019 edition ed.). Springer, New York, NY, USA.

[53] Lene Nielsen, Kira Storgaard Hansen, Jan Stage, and Jane Billestrup. 2015. A Template for Design Personas: Analysis of 47 Persona Descriptions from Danish Industries and Organizations. *International Journal of Sociotechnology and*

Knowledge Development 7, 1 (Jan. 2015), 45–61. https://doi.org/10.4018/ijskd.2015010104

[54] Lene Nielsen, Kira Storgaard Nielsen, Jan Stage, and Jane Billestrup. 2013. Going Global with Personas. In *Proceedings of the INTERACT 2013 conference* (2013). Springer, Berlin, Heidelberg, Cape Town, South Africa, 350–357. https://doi.org/10.1007/978-3-642-40498-6_27

[55] College of Information and Computer Sciences. 2020. EQUATE. https://groups.cs.umass.edu/equate/

[56] Pentti Paatero and Unto Tapper. 1994. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics* 5, 2 (1994), 111–126. https://doi.org/10.1002/env.3170050203

[57] John Pruitt and Jonathan Grudin. 2003. Personas: Practice and Theory *(DUX '03)*. ACM, San Francisco, California, USA, 1–15. https://doi.org/10.1145/997078.997089

[58] Jing Qian and Venkatesh Saligrama. 2013. Spectral Clustering with Unbalanced Data. *arXiv:1302.5134 [stat]* (Feb. 2013). http://arxiv.org/abs/1302.5134 arXiv: 1302.5134.

[59] Danijel Radjenović, Marjan Heričko, Richard Torkar, and Aleš Živkovič. 2013. Software fault prediction metrics: A systematic literature review. *Information and software technology* 55, 8 (2013), 1397–1418.

[60] Joni Salminen, Willemien Froneman, Soon-gyo Jung, Shammur Chowdhury, and Bernard J. Jansen. 2020. The Ethics of Data-Driven Personas. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems Extended Abstracts (CHI '20)*. Association for Computing Machinery, Honolulu, HI, USA, 1–9. https://doi.org/10.1145/3334480.3382790

[61] Joni Salminen, Kathleen Guan, Soon-Gyo Jung, Shammur Absar Chowdhury, and Bernard J. Jansen. 2020. A Literature Review of Quantitative Persona Creation. In *Proceedings of the ACM Conference of Human Factors in Computing Systems (CHI'20)* (2020). ACM, Honolulu, Hawaii, USA. https://doi.org/10.1145/3313831.3376502

[62] Joni Salminen, Bernard J. Jansen, Jisun An, Haewoon Kwak, and Soon-gyo Jung. 2018. Are personas done? Evaluating their usefulness in the age of digital analytics. *Persona Studies* 4, 2 (Nov. 2018), 47–65. https://doi.org/10.21153/psj2018vol4no2art737

[63] Joni Salminen, Joao M. Santos, Haewoon Kwak, Jisun An, Soon-gyo Jung, and Bernard J. Jansen. 2020. Persona Perception Scale: Development and Exploratory Validation of an Instrument for Evaluating Individuals' Perceptions of Personas. *International Journal of Human-Computer Studies* (April 2020), 102437. https://doi.org/10.1016/j.ijhcs.2020.102437

[64] Joni Salminen, Jukka Vahlo, Aki Koponen, Soon-Gyo Jung, Shammur A. Chowdhury, and Bernard J. Jansen. 2020. Designing Prototype Player Personas from a Game Preference Survey. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems Extended Abstracts (CHI '20)*. Association for Computing Machinery, Honolulu, HI, USA, 1–8. https://doi.org/10.1145/3334480.3382785

[65] Joni Salminen, Sercan Şengün, Haewoon Kwak, Bernard J. Jansen, Jisun An, Soon-gyo Jung, Sarah Vieweg, and Fox Harrell. 2017. Generating Cultural Personas from Social Data: A Perspective of Middle Eastern Users. In *Proceedings of The Fourth International Symposium on Social Networks Analysis, Management and Security (SNAMS-2017)*. IEEE, Prague, Czech Republic. https://doi.org/10.1109/FiCloudW.2017.97

[66] Joni Salminen, Sercan Şengün, Haewoon Kwak, Bernard J. Jansen, Jisun An, Soon-gyo Jung, Sarah Vieweg, and Fox Harrell. 2018. From 2,772 segments to five personas: Summarizing a diverse online audience by generating culturally adapted personas. *First Monday* 23, 6 (June 2018). https://doi.org/10.5210/fm.v23i6.8415

[67] Amit Saxena, Mukesh Prasad, Akshansh Gupta, Neha Bharill, Om Prakash Patel, Aruna Tiwari, Meng Joo Er, Weiping Ding, and Chin-Teng Lin. 2017. A review of clustering techniques and developments. *Neurocomputing* 267 (Dec. 2017), 664–681. https://doi.org/10.1016/j.neucom.2017.06.053

[68] David A. Siegel. 2010. The Mystique of Numbers: Belief in Quantitative Approaches to Segmentation and Persona Development. In *CHI '10 Extended Abstracts on Human Factors in Computing Systems (CHI EA '10)*. ACM, New York, NY, USA, 4721–4732. https://doi.org/10.1145/1753846.1754221 event-place: Atlanta, Georgia, USA.

[69] Rashmi Sinha. 2003. Persona Development for Information-rich Domains. 830–831. https://doi.org/10.1145/765891.766017

[70] Brian A. Smith and Shree K. Nayar. 2016. Mining Controller Inputs to Understand Gameplay. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. Association for Computing Machinery, Tokyo, Japan, 157–168. https://doi.org/10.1145/2984511.2984543

[71] Phillip Douglas Stevenson and Christopher Andrew Mattson. 2019. The Personification of Big Data. *Proceedings of the Design Society: International Conference on Engineering Design* 1, 1 (July 2019), 4019–4028. https://doi.org/10.1017/dsi.2019.409

[72] Molly L. Tanenbaum, Rebecca N. Adams, Esti Iturralde, Sarah J. Hanes, Regan C. Barley, Diana Naranjo, and Korey K. Hood. 2018. From Wary Wearers to d-Embracers: Personas of Readiness to Use Diabetes Devices. *Journal of Diabetes Science and Technology* 12, 6 (Nov. 2018), 1101–1107. https://doi.org/10.1177/1932296818793756

[73] N. Tu, X. Dong, P. P. Rau, and T. Zhang. 2010. Using cluster analysis in Persona development. In *2010 8th International Conference on Supply Chain Management and Information*. 1–5.

[74] Phil Turner and Susan Turner. 2011. Is stereotyping inevitable when designing with personas? *Design studies* 32, 1 (2011), 30–44.

[75] S. Vosbergen, J. M. R. Mulder-Wiggers, J. P. Lacroix, H. M. C. Kemps, R. A. Kraaijenhagen, M. W. M. Jaspers, and N. Peek. 2015. Using personas to tailor educational messages to the preferences of coronary heart disease patients. *Journal of Biomedical Informatics* 53 (Feb. 2015), 100–112. https://doi.org/10.1016/j.jbi.2014.09.004

[76] L. Wang, L. Li, H. Cai, L. Xu, B. Xu, and L. Jiang. 2018. Analysis of Regional Group Health Persona Based on Image Recognition. In *2018 Sixth International Conference on Enterprise Systems (ES)*. 166–171. https://doi.org/10.1109/ES.2018.00033

[77] Yasuhiro Watanabe, Hironori Washizaki, Kiyoshi Honda, Yuki Noyori, Yoshiaki Fukazawa, Aoi Morizuki, Hiroyuki Shibata, Kentaro Ogawa, Mikako Ishigaki, Satiyo Shiizaki, Teppei Yamaguchi, and Tomoaki Yagi. 2017. ID3P: Iterative Data-driven Development of Persona Based on Quantitative Evaluation and Revision. In *Proceedings of the 10th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE '17)*. IEEE Press, Piscataway, NJ, USA, 49–55. https://doi.org/10.1109/CHASE.2017.9 event-place: Buenos Aires, Argentina.

[78] Mark D. Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E. Bourne, Jildau Bouwman, Anthony J. Brookes, Tim Clark, Mercè Crosas, Ingrid Dillo, Olivier Dumon, Scott Edmunds, Chris T. Evelo, Richard Finkers, Alejandra Gonzalez-Beltran, Alasdair J. G. Gray, Paul Groth, Carole Goble, Jeffrey S. Grethe, Jaap Heringa, Peter A. C. 't Hoen, Rob Hooft, Tobias Kuhn, Ruben Kok, Joost Kok, Scott J. Lusher, Maryann E. Martone, Albert Mons, Abel L. Packer, Bengt Persson, Philippe Rocca-Serra, Marco Roos, Rene van Schaik, Susanna-Assunta Sansone, Erik Schultes, Thierry Sengstag, Ted Slater, George Strawn, Morris A. Swertz, Mark Thompson, Johan van der Lei, Erik van Mulligen, Jan Velterop, Andra Waagmeester, Peter Wittenburg, Katherine Wolstencroft, Jun Zhao, and Barend Mons. 2016. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data* 3, 1 (March 2016), 160018. https://doi.org/10.1038/sdata.2016.18 Number: 1 Publisher: Nature Publishing Group.

[79] Jeannette M. Wing. 2020. Data for Good: FATES, Elaborated. https://datascience.columbia.edu/FATES-Elaborated

[80] Svante Wold, Kim Esbensen, and Paul Geladi. 1987. Principal component analysis. *Chemometrics and Intelligent Laboratory Systems* 2, 1 (Aug. 1987), 37–52. https://doi.org/10.1016/0169-7439(87)80084-9

[81] Bernhard Wöckl, Ulcay Yildizoglu, Isabella Buber, Belinda Aparicio Diaz, Ernst Kruijff, and Manfred Tscheligi. 2012. Basic Senior Personas: A Representative Design Tool Covering the Spectrum of European Older Adults. In *Proceedings of the 14th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '12)*. ACM, New York, NY, USA, 25–32. https://doi.org/10.1145/2384916.2384922 event-place: Boulder, Colorado, USA.

[82] Jing-Hao Xue and D. Titterington. 2008. Do unbalanced data have a negative effect on LDA? *Pattern Recognition* 41 (May 2008), 1558–1571. https://doi.org/10.1016/j.patcog.2007.11.008

[83] Tetsuya Yoshida. 2013. Learning and Utilizing a Pool of Features in Non-negative Matrix Factorization. In *Active Media Technology (Lecture Notes in Computer Science)*, Tetsuya Yoshida, Gang Kou, Andrzej Skowron, Jiannong Cao, Hakim Hacid, and Ning Zhong (Eds.). Springer International Publishing, Cham, 96–105. https://doi.org/10.1007/978-3-319-02750-0_10

[84] Xiang Zhang, Hans-Frederick Brown, and Anil Shankar. 2016. Data-driven Personas: Constructing Archetypal Users with Clickstreams and User Telemetry. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (2016) *(CHI '16)*. ACM, San Jose, California, USA, 5350–5359.

[85] Haining Zhu, Hongjian Wang, and John M. Carroll. 2019. Creating Persona Skeletons from Imbalanced Datasets - A Case Study using U.S. Older Adults' Health Data. In *Proceedings of the 2019 on Designing Interactive Systems Conference - DIS '19*. ACM Press, San Diego, CA, USA, 61–70. https://doi.org/10.1145/3322276.3322285

APPENDIX 1: THE IMPLEMENTATION OF THE ALGORITHMS

Cluster Analysis

CA is often considered as the default technique for exploring similarity within multidimensional data [67]. CA is a collection of unsupervised learning methods that aim to group objects depending on how similar they are. There are several CA techniques that can be divided into hierarchical clustering (HC) and partitional clustering (PA).

HC groups individual items into larger groups by computing distances between different elements to produce clusters in a hierarchical order based on similarity. The clustering procedure continues until the data is reduced to the number of clusters set as the hyperparameter.

PA breaks down larger objects into smaller ones. In contrast to HC, no hierarchy is followed when forming the clusters. K-means is a typical PC method, where the user defines the number of

clusters into which the data are divided. These clusters reveal common patterns within distinct groups and can be built upon to form personas [61].

For our implementation of clustering, we use the Feature Agglomeration from the Python library of `Scikit-learn`[6], which corresponds to the clustering type of HC. The parameters of the algorithm are as follows:

- **n_clusters:** Number of clusters to create. We used 5, 10, and 15.
- **affinity:** Metric used to compute the association between the clusters. We used 'euclidean' (default). Other options are 'l1', 'l2', 'manhattan', 'cosine', and 'precomputed'.
- **linkage:** The linkage parameter decides which distance to use between sets of features. The algorithm tries to merge the pairs that minimize this criterion. We used the 'Ward' (default option) that minimizes the variance of the clusters. Other options include 'Average' (averaging the distance of each feature), 'Complete' (using the maximum distances between all the features), and 'Single' (using the minimum distance between all observations).

Principal Component Analysis

PCA is one of the most widely used dimensionality reduction (DR) techniques aimed at dealing with a large number of numerical values [80]. PCA is considered easy to use; it works for the vast majority of datasets and is offered in most data science packages alongside with supervised methods (partial least square discriminate analyses). PCA uses a matrix of numerical values as an input, denoted by N rows (objects) and K columns (variables).

Based on this input, PCA performs eigenvalue decomposition over the covariance matrix. The eigenvector matrix is then used to project the samples into a new subspace, denoted by l vectors named principal components (PC), where l is any number less than the original dimension, K. PCA also estimates how much variance of the data is captured by its PCs, using $Q2$ that represents the residual variance within the data left out of the model.

The disadvantage of PCA is that it is hard to interpret, and in many instances, it requires manual fine-tuning. For instance, PCA is greatly affected by the type of scaling used for the analysis and can yield completely different results using different scaling parameters. In general, PCA does not perform well with imbalanced datasets, as these can introduce artifacts in the analysis.

For our study, we use the Scikit-learn implementation of PCA[7]. The parameters of the algorithm are as follows:

- **n_components:** Number of segments to keep. We used 5, 10, and 15.

Non-negative Matrix Factorization

NMF is an unsupervised method employing matrix factorization [41], i.e., decomposition of a matrix into two or more matrices in which all values are equal to or greater than zero [17]. The calculations behind NMF are similar to PCA but aim to explain all of the contents in the data matrix, not just their correlations. However, when PCA is used without the non-negativity constraints, the two methods are equal in performance [56]. NMF is particularly useful when dealing with real-world datasets that are often non-negative. For example, user interaction data and user characteristics are almost always equal or greater than zero. NMF, similarly to PCA, has challenges with using imbalanced data or data with a large number of features [83].

For our study, we use the Scikit-learn implementation of NMF[8]. The parameters of the algorithm are as follows:

[6]https://scikit-learn.org/stable/modules/generated/sklearn.cluster.FeatureAgglomeration.html
[7]https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html
[8]https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.NMF.html

- **n_components:** Number of segments to keep. We used 5, 10, and 15.

Latent Dirichlet Allocation

LDA is an unsupervised probabilistic topic model applied to process large amounts of digital data. LDA applies a generative probabilistic model commonly used for inferring latent patterns (topics), typically from text. When processing a dataset, LDA returns a vector representing each instance's probability associated with each latent component. LDA deals particularly well with imbalanced datasets, and the re-balancing of datasets has been explored, concluding that imbalanced data do not affect LDA performance [82].

For our study, we use the Scikit-learn implementation of LDA[9]. The parameters of the algorithm are as follows:

- **n_components:** Number of segments to keep. We used 5, 10, and 15.

Spectral Embedding

SE uses a similarity graph to perform non-linear DR [43]. It forms an affinity matrix, and using spectral decomposition, returns a transformation based on the eigenvectors for each data point. SE is particularly sensitive to imbalanced datasets, which can lead to poor performance on graphs [58].

For our study, we use the Scikit-learn implementation of SE[10]. The parameters of the algorithm are as follows:

- **n_components:** Number of segments to keep. We used 5, 10, and 15.
- **affinity:** There are 5 types to construct an affinity matrix. We use the 'nearest_neighbors' that computes a nearest neighbors graph and constructs the affinity matrix. Other options are 'Rbf' (uses Radial Basis function to construct affinity matrix), 'Precomputed' (interprets X as a precomputed affinity matrix), and 'precomputed_nearest_neighbors' (first interprets X as a sparse graph of precomputed nearest neighbors and then constructs the affinity matrix by selecting the nearest neighbors).

Uniform Manifold Approximation and Projection

UMAP is a non-linear DR technique [47]. It is very close to t-distributed stochastic neighbor embedding (tSNE), developed for effective two or three-dimensional data visualizations [44]. UMAP, similar to SE, is sensitive to imbalanced datasets. This is because both SE and UMAP use the n neighbors parameter for locating the amount of nearest neighbors used to construct the high-dimensional graph. Setting the value of n neighbors is a balancing act between preserving local versus global structure in the data, fine-grained versus large-scale features [47]. A low value will push UMAP towards focusing on the local structure, while a high value will preserve global structure in the projection, but losing the fine detail.

For our study, we use the Umap-learn library[11]. The parameters of the algorithm are as follows:

- **n_components:** Number of segments to keep. We used 5, 10, and 15.

APPENDIX 2: CONSISTENCY SCORE: GENERAL CASE

Let us consider the general case of *CS*, where both the number of the persona sets and the number of personas in each set are hyperparameters.

[9]https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.LatentDirichletAllocation.html
[10]https://scikit-learn.org/stable/modules/generated/sklearn.manifold.SpectralEmbedding.html
[11](https://github.com/lmcinnes/umap)

For example, in the case, we have four sets of personas, we need to choose two in four for each comparison of consistency. Thus there are $\binom{4}{2}$ combinations = $\frac{4!}{2!2!}$ = 6 combinations. We will have six *CS* values, and then we take the average to compute the final *CS* of an algorithm.

In the case we have M sets of personas, we need to choose two in M for each time for comparison of consistency. Thus there are $\binom{M}{2}$ combinations = $\frac{M!}{2!(M-2)!} = \frac{M*(M-1)}{2}$ combinations. We will have $\frac{M*(M-1)}{2}$ *CS* values and then we take their average.

Suppose that we have m set of personas denoted M_1 to M_m. n_{M_1} is the number of personas in set M_1, n_{M_2} is the number of personas in set M_2, ... n_{M_m} is the number of personas in set M_m.

Without loss of generality, we set $n_{M_1} <= n_{M_2} <= \ldots <= n_{M_m}$.

Then, the *CS* is calculated as:

$$\frac{1}{\binom{M}{2}} \sum_{i=1}^{m-1} \left(\sum_{k=i+1}^{m} \left(\frac{\sum_{j=1}^{n_{M_i}} \mathbb{1}_{P_{M_{ij}} \in \{P_{M_k}\}}}{n_{M_i}} \right) \right)$$

$$= \frac{1}{M*(M-1)/2} \sum_{i=1}^{m-1} \left(\sum_{k=i+1}^{m} \left(\frac{\sum_{j=1}^{n_{M_i}} \mathbb{1}_{P_{M_{ij}} \in \{P_{M_k}\}}}{n_{M_i}} \right) \right),$$

where $P_{M_{ij}}$ denotes the persona j of set M_i and $\{P_{M_k}\}$ denote all the personas of set M_k.