ChatGPT

Proiectarea generală a sistemului

Vom crea un **pipeline complet** pentru generarea a 100.000 de **personaje sintetice realiste și coerente** („persoane") reprezentative pentru populația Republicii Moldova, folosind datele **Recensământului național din 2024** disponibile prin platforma PxWeb. Scopul este obținerea unui set de date sintetic, **în limba română**, potrivit pentru antrenarea și simularea modelelor AI – analog dataset-ului de *personas* al NVIDIA (Nemotron-Personas) – dar adaptat la demografia reală a Moldovei. Vom descrie schema datelor, relațiile și distribuțiile folosite la eșantionare, metodele de generare (PGM/IPF), șabloanele de prompturi folosite pentru generarea narațiunilor, mecanisme de validare a coerenței și corectitudinii statistice, precum și pașii finali de ambalare (Parquet, metadata, licență, documentație). Importanța majoră este ca distribuțiile **să reflecte fidel structura reală a populației** (fără supra-eșantionarea unor subgrupuri minoritare) [1] [2].

Schema recomandată a dataset-ului

Vom utiliza o schemă mixtă, cu **câmpuri structurate (tabulare)** și **câmpuri narative** (text liber generat de model lingvistic). Schema urmărește modelul celor 22 de câmpuri din dataset-ul NVIDIA Nemotron-Personas [3], adaptat la contextul Republicii Moldova. Se vor include aproximativ **6 câmpuri narative** principale și circa **15-16 câmpuri contextuale/structurate**, după cum urmează:

- **ID unic (`uuid`)** – Identificator unic (de exemplu un UUID hexazecimal de 32 caractere).
- **Nume și prenume (`name`)** – Numele fictiv al persoanei (genereat conform distribuției etno-lingvistice).
- **Sex (`sex`)** – Genul biologic, valori posibile: `Masculin` sau `Feminin`. Distribuția va fi ~47% masculin, ~53% feminin conform recensământului [4].
- **Vârstă (`age`)** – Vârsta în ani (întregi). Intervalul 0 – 100+ ani (distribuție bazată pe piramida populației; medie ~40,6 ani [5]). Vom asigura proporțiile reale pe grupe de vârstă: ~19,2% sub 15 ani, ~62,7% între 15-64 ani, ~18,1% 65+ ani [6].
- **Stare civilă (`marital_status`)** – De exemplu: `necăsătorit/necăsătorită`, `căsătorit(ă)`, `divorțat(ă)`, `văduv(ă)`, `separat(ă)` etc. Distribuția pentru populația 15+ va ține cont de cifrele recensământului (ex: ~55,8% căsătoriți, ~23,6% necăsătoriți, ~10% văduvi, ~10% divorțați [7]), ajustat pe grupe de vârstă (vârste tinere predominant necăsătoriți, vârste înaintate cu pondere mai mare de văduvi etc).
- **Nivel educațional (`education_level`)** – Cel mai înalt nivel absolvit (conform clasificării ISCED adaptate): de ex. `fără studii`, `primar`, `gimnazial`, `liceal`, `profesional/tehnic` (școală profesională sau postliceală), `universitar` (licență/master) sau `doctorat`. Distribuțiile se vor extrage din datele recensământului: pondere ~19,1% studii superioare (licență/master) [8], ~33,6% studii profesionale tehnice [9], ~12,8% liceal, ~22,7% gimnazial, ~9,2% primar, ~2,4% fără școală (valorile exacte vor fi calibrate pentru populația de 10 ani și peste [10]).
- **Domeniu specializare (`field_of_study`)** – *[Opțional]* Pentru persoanele cu studii superioare, putem include un câmp cu domeniul general al studiilor (ex. `STEM`, `Științe sociale`, `Economie`, `Arte/Uman`, `Educație`, `Sănătate` etc., conform distribuirii absolvenților pe domenii). Dacă astfel de date nu sunt direct disponibile din recensământ, se pot folosi surse educaționale naționale sau internaționale (ex. pondere absolvenți pe domenii) pentru aproximare. **Notă:** Acest câmp nu exista explicit în dataset-ul Nemotron-Personas USA (acolo era

`bachelors_field` cu 6 valori) [11], dar îl putem include pentru bogăția contextului, mai ales dacă datele permit.

- **Ocupație (`occupation`)** – Profesia sau ocupația actuală a persoanei. Vom folosi o listă detaliată de ocupații (ideal peste 500 de categorii, similar cu cele ~567 ocupații distincte din dataset-ul Nvidia [12]). Lista poate fi derivată din Clasificatorul Ocupațiilor din RM (COR) sau din nomenclatorul ISCO-08, adaptată. Distribuția ocupațiilor va fi *condiționată* de alte variabile (ex: vârstă, sex, educație, mediul urban/rural) astfel încât să reflecte piața muncii din RM. De exemplu, ponderi mai mari pentru agricultură în rural, joburi în IT concentrate în mediul urban (Chișinău), etc. Ocupația `none` (şomer, copil sau pensionar) va fi atribuită persoanelor care nu lucrează (minori, vârstnici peste limita de pensionare sau alte cazuri). Pentru elevi/studenți, putem trece `elev` sau `student` ca ocupație.
- **Locație: localitate și regiune (`city/town/village` și `region`)** – Localitatea de reședință (nume de oraș sau sat) și regiunea (raionul sau unitatea administrativ-teritorială de nivel II, ori regiunea de dezvoltare). Vom genera localități fictive, dar realiste, alegând din lista celor ~1500 localități recenzate [13], sau putem folosi numai *denumiri reale* pentru credibilitate (ex.: Chișinău, Bălți, Comrat, s. Selemet, etc.). **Regiunea de dezvoltare** (cele 5 regiuni: Chișinău, Centru, Nord, Sud, UTA Găgăuzia) va fi derivată din localitate. Distribuția pe regiuni în setul sintetic va respecta strict ponderea populației reale: ~29,9% Chișinău, ~27,8% Centru, ~25,3% Nord, ~12,7% Sud, ~4,3% Găgăuzia [2]. De asemenea, vom asigura și echilibrul **urban/rural**: ~46,4% urban, 53,6% rural [4] (marcat eventual printr-un câmp separat `residence_type`: Urban/Rural).
- **Fundal cultural (`cultural_background`)** – Câmp narativ scurt care descrie originile etno-culturale, limba maternă și contextul familial al persoanei. Acesta va fi generat de modelul LLM pe baza etniei și religiei declarate, a regiunii de proveniență etc. (ex.: „Ion este originar dintr-o familie de *găgăuzi* din sudul Moldovei, crescând într-un sat multietnic unde se vorbesc atât *găgăuza*, cât și *româna*, într-o comunitate ortodoxă unită”). **Notă:** În dataset-ul original Nemotron, un câmp similar `cultural_background` furniza detalii despre originea etnică și mediul în care a crescut persona [14].
- **Narațiuni de personalitate (personas narrative)** – Vom genera **5-6 paragrafe narative** distincte, care împreună conturează personalitatea și viața personajului. Urmărim formatul din Nemotron-Personas, adaptat în română:
- `descriere_generala` – un paragraf despre trăsăturile definitorii de personalitate ale individului, stilul său de viață și comportament (ex.: „este o fire sociabilă și organizată, cu un simț pronunțat al responsabilității, mereu dornică să învețe lucruri noi...”).
- `profil_profesional` – descrierea vieții profesionale: ocupația, abilitățile la locul de muncă, etica profesională, ambițiile de carieră etc. (ex.: „lucrează ca inginer constructor, cunoscut pentru atenția la detalii și abilitatea de a coordona echipe; visează să devină manager de proiect...”).
- `hobby_sport` – interese sportive, activități fizice, rutină de sănătate (ex.: „în timpul liber aleargă în parc și joacă fotbal la sfârșit de săptămână cu prietenii, fiind un suporter înfocat al echipei naționale...”).
- `hobby_arta_cultura` – gusturi artistice și culturale: muzică preferată, literatură, filme, hobby-uri creative (ex.: „iubește muzica populară moldovenească pe care a învățat-o de la bunici, citește romane istorice și participă la festivaluri locale de teatru...”).
- `hobby_calatorii` – preferințe legate de călătorii și vacanțe: dacă îi place să călătorească, unde, stil de călătorie (ex.: „îi plac escapadele scurte la munte în România și visează să viziteze Italia pentru patrimoniul cultural bogat; își planifică riguros bugetul de călătorie...”).
- `hobby_culinar` – obiceiuri culinare și aptitudini gastronomice: ce gătește sau ce mănâncă, bucătăria preferată (ex.: „gătește adesea bucate tradiționale moldovenești ca plăcintele și sarmalele după rețeta mamei, dar experimentează și rețete internaționale; e pasionat de vinurile locale...”).

Fiecare dintre aceste secțiuni va fi generată ca un paragraf coerent în română, de 2-5 propoziții, integrând detalii relevante despre persoană. Vom stoca aceste paragrafe ca câmpuri text separate (similar celor 6 fields narative din setul US: `persona`, `professional_persona`, `sports_persona`, `arts_persona`, `travel_persona`, `culinary_persona` [15]). În final, concatenarea lor (în ordinea logică) ar reda un portret narativ complex al persoanei, comparabil cu exemplarele Nemotron (vezi exemplul cu Mary Alberti [16] [17]). - **Abilități și interese (liste)** – În completarea textelor narative, vom include și versiuni structurate (listă/array) extrase din acestea, pentru a facilita filtrarea sau analiza atributelor cheie: - `skills_and_expertise_list` – listă de competențe profesionale cheie, extrase din textul profilului profesional (ex.: `["management financiar", "negociere", "gestionare proiecte"]`). - `hobbies_and_interests_list` – listă de hobby-uri principale, extrase din paragrafele de hobby (ex.: `["fotbal", "lectură", "grădinărit"]`). - (Opțional, dacă avem nevoie de consistență cu dataset-ul original, putem include și variante textuale combinate ale acestor liste: câmpurile `skills_and_expertise` și `hobbies_and_interests` ca fraze descriptive, însă acestea sunt în esență redundante față de paragrafele deja generate.) - **Câmpuri geografice suplimentare** – Pentru conformitate cu formatul Parquet al Nemotron, putem include: - `district` – raionul (sau municipiul) de reședință, dacă decidem să folosim această unitate ca nivel în loc de regiunea de dezvoltare. Ar avea ~35 valori (inclusiv UTA și mun. Chișinău/Bălți). - `country` – țara, în mod constant `Moldova` (echivalent cu `country = USA` din dataset-ul original [18]).

Notă: Unele câmpuri ar putea fi omise sau combinate pentru eficiență. De exemplu, dacă includem raionul (`district`), regiunea de dezvoltare se poate deduce, deci `region` devine redundant – dar îl putem păstra pentru filtrări rapide. În mod similar, `residence_type` (urban/rural) se deduce din localitate sau raion, însă un câmp explicit poate fi convenabil. Scopul este să facem schema *cât mai informativă*, aliniată la practica Nemotron, dar adaptată la datele disponibile pentru Moldova.

Graful de dependențe și relațiile de eșantionare

Pentru a **produce date structurate realiste**, vom defini un **graf de dependențe** între variabile, astfel încât eșantionarea să respecte nu doar distribuțiile marginale, ci și corelațiile demografice importante. În loc să asumăm independența câmpurilor, vom utiliza relații condiționate extrase din tabelele recensământului (prin PxWeb). Graful de dependențe poate fi conceptualizat astfel (variabilele de bază influențează variabilele derivate):

```
[Regiune, Urban/Rural] ──► [Etnie, Limbă maternă, Religie]
                  │          (compoziție diferită pe zone)
                  │
                  ├──► [Sex] (aprox. independent de regiune la nivel macro)
                  │
                  ├──► [Vârstă] ──► [Stare civilă]
                  │         └──► [Educație]
                  │                   └──► [Ocupație]
                  │                         └──► [Venit estimat]* (dacă s-ar
include)
                  │
                  └──► [Localitate concretă]
```

(diagrama conceptuală a relațiilor; săgețile indică influența/condiționarea unei variabile de alta)

Explicații ale relațiilor cheie:

- **Regiune de dezvoltare și mediul (urban/rural)**: Acestea sunt variabile de nivel macro stabilite primele, deoarece determină multe alte caracteristici. Distribuția pe regiuni și mediul urban/rural este cunoscută și fixă conform datelor oficiale (ex: 46,4% urban [4], cu ~30% din total în Chișinău [2]). Regiunea și mediul vor influența:
- **Etnie și limbă**: Populația Moldovei este majoritar moldovenească/română (~77,2% s-au declarat moldoveni, 7,9% români la recensământul 2024 [19]). Însă distribuția minorităților depinde geografic. De exemplu, **în Găgăuzia 81,9% din populație s-a declarat găgăuză** și doar ~18% moldoveni, pe când în Centru peste 87% sunt moldoveni [20]. Vom modela probabilitatea etniei condiționat de regiune (ex: dacă regiune=Găgăuzia, P(etnie=Găgăuz) ≈0,82; dacă regiune=Sud și raion=Taraclia, P(etnie=Bulgar) ridicată șamd.). Limba maternă se corelează cu etnia (ex: cei declarați ruși/găgăuzi e probabil să aibă limba maternă rusă/găgăuză; ~11,1% din pop. a declarat rusa ca limbă maternă, similar procentului combinat de ruși+ucraineni [21]). Vom folosi distribuțiile etno-lingvistice per regiune pentru consistență.
- **Religie**: Majoritatea populației este ortodoxă (~95,0% [22]). Vom menține această pondere ridicată, introducând totodată minorități confesionale (baptiști ~1,1%, martori ~0,7%, penticostali ~0,5% etc, plus 0,6% atei [22]). Acestea nu variază extrem pe regiuni, dar ruralul poate avea ușor mai mulți practicanți conservatori. Putem considera religia aproape independentă de restul (sau corelată slab cu etnia).
- **Sex**: Vom trata sexul ca variabilă independentă (repartizată ~47% masculin, ~53% feminin [4]) și, dacă e necesar, vom asigura consistență la nivel micro (de ex., în cupluri căsătorite generate putem verifica număr relativ egal bărbați/femei, însă pentru setul de personas individuale, nu e necesară legarea explicită). În unele regiuni pot exista ușoare deviații (ex.: în Chișinău femeile pot fi >53% datorită migrației, iar în mediul rural bărbații ceva mai puțini din cauza mortalității/ exodului), dar ne putem rezuma la distribuția generală pe țară.
- **Vârstă**: Se va eșantiona din distribuția pe vârste a populației (posibil pe an de naștere sau pe grupe fine). Pentru acuratețe, vom folosi datele detaliate ale piramidei populației 2024 (disponibile prin PxWeb – tabele de populație rezidentă pe sex și vârstă). De exemplu, vom asigura că ~18,1% sunt vârstnici 65+ [6], iar vârful populației e în jurul cohortelor 30-40 ani (populație îmbătrânită). **Vârsta influențează puternic**:
- **Starea civilă**: Vom aplica distribuții condiționate pe grupe de vârstă și sex. De ex., la 20-24 ani majoritatea sunt necăsătoriți; la 30-50 ani predominant căsătoriți; la 70+ ani multe văduve (femeile trăiesc mai mult). Avem date de la recensământ privind structura maritală: 55,8% căsătoriți din total ≥15 ani [7], dar vom detalia: pondere „necăsătorit" scade cu vârsta, „divorțat" crește la generația mijlocie etc. Putem folosi *tabele de contingenta* (ex: căsătorit/ necăsătorit pe grupe de vârstă și sex) dacă sunt publicate, sau date de stare civilă anterioare calibrate la noile margini.
- **Nivelul educațional**: Puternic corelat cu vârsta (persoanele în vârstă au în medie studii mai scăzute, tinerii au șanse mai mari la studii superioare decât acum 50 de ani). Vom reflecta această evoluție: ex. generația 20-35 are cea mai mare rată de studii superioare, conform datelor (a crescut la 19,1% per total ≥10 ani [8], și chiar mai ridicată la tineri; femeile tinere ~39% cu studii superioare [23]). Vom integra date din comunicatele BNS [24] [10]: ~33,6% au studii profesionale tehnice per total, ceea ce sugerează majoritar pentru generațiile de mijloc, etc. Putem construi distribuții condiționate pe generații: de ex. <25 ani – mulți încă „în studii" (20,6% continuau studiile la momentul recensământului [8], deci la 18-22 ani vom marca mulți ca „student/în curs de educație"), 25-35 ani – vârf absolvenți de universitate, >60 ani – cel mai probabil nivel gimnazial sau profesional.
- **Educație ↦ Ocupație**: Nivelul de educație va influența tipul de ocupație generat:
- Persoanele cu studii superioare vor fi plasate preponderent în ocupații de *specialiști*, *funcționari*, *manageri*, *lucrători în birouri*, etc. (conform distribuției ocupațiilor pe niveluri de calificare).

- Persoanele cu studii medii profesionale pot fi tehnicieni, meșteșugari, șoferi, operatori, lucrători calificați în diverse domenii tehnice.
- Persoanele cu studii gimnaziale sau primare vor fi fie muncitori necalificați (agricultură, construcții necalificate, personal de curățenie), fie nu vor avea un job formal (șomeri, casnici).
- De asemenea, **vârsta** se combină cu educația: tinerii sub 18 ani vor avea ocupația „elev/student" (dacă <16 ani, oblig. elevi; 16-22 mulți studenți), vârstnicii peste ~63 ani probabil pensionari (dacă includem un status, altfel putem genera totuși o ocupație anterioră sau hobby drept ocupație; pentru realism, e mai bine să marcăm lipsa ocupației sau „pensionar, fost…"). Vom programa ca dacă `age > 65` și nu e altceva, atunci ocupație = „pensionar (fostă profesie: X)" și putem incorpora asta în narațiune.
- **Locația** influențează ocupația: din datele economice știm că în rural predomină agricultura (fermier, muncitor agricol), pe când în Chișinău servicii și industrie ușoară. Vom folosi tabele de ocupare pe județe/raioane sau macro (ex.: structură ocupațională pe urban/rural). Dacă nu avem direct din recensământ (care totuși a colectat ocupația și poate BNS a publicat indicatori, ex: % populație ocupată în agricultură vs servicii), putem folosi datele Inspecției Muncii sau sondaje. Important este să evităm alocarea unei ocupații improbabile: ex. un IT-ist în sat izolat fără internet – foarte puțin probabil. Deci vom condiționa: dacă mediul = rural, probabilitățile pentru „agricultor, crescător de animale, învățător, preot, mic comerciant" cresc, iar pentru „programator, manager financiar" scad drastic (dar nu zero dacă persoana face naveta sau lucrează remote, caz rar).
- **Etnie/Limbă ↦ Nume și context cultural**: Pe baza etniei alese (condiționat de regiune cum am spus), vom genera **numele** persoanei corespunzător. Vom pregăti liste de prenume și nume de familie frecvente pentru fiecare grup etnic/l Lingvistic:
- Moldoveni/Români: prenume ca *Ion, Maria, Elena, Vasile, Ana, Ștefan, Daniela,* etc.; nume de familie tipice: *Popa, Rusu, Sandu, Ciobanu, etc.* (Se pot folosi date de la evidența populației sau liste onomastice publice).
- Ruși/Ucraineni: prenume *Ivan, Olga, Tatiana, Mihail, etc.*; nume *Ivanov, Bodiul, etc.*
- Găgăuzi: prenume turcice sau slavonizate *Maria (comun), Ivan, Dimitri, Ana, etc.* și nume găgăuze (des întâlnite: *Topal, Esir, Jardan,* etc.).
- Bulgari: *Ivan, Stoian, Elena,* nume precum *Petrov, Georgiev* etc.
- Roma: nume specifice dacă dorim, sau pot avea nume românești comune.
- Vom asigura ca ~77% din personas au nume românești (considerând moldoveni+români împreună) [19] , ~5% ucrainene, ~4% găgăuze, ~3% rusești, ~1.6% bulgărești, etc., conform structurii etnice [19] . Totodată, vom integra *limba preferată* în narațiune: ex. un găgăuz va avea în paragraf cultural mențiuni despre limba găgăuză și poate bilingvism rus/român, un rus din Chișinău menționează că vorbește preponderent rusa, etc. Astfel, *fără a avea un câmp structurat separat pentru limbă*, reflectăm realitatea lingvistică (49% au declarat limba „moldovenească" drept maternă și 31% româna – împreună ~80% practic limba română – restul rusă 11%, găgăuză 3.8%, ucraineană 2.9% etc [21]).
- **Alte dependențe**: Ocupația poate influența *venitul* sau *statutul socioeconomic* – dacă am include astfel de atribute. Deocamdată nu sunt cerute, dar narațiunile pot implica statut (de ex. un manager urban are un trai mai prosper vs. un muncitor agricol modest). Modelul LLM poate deduce tonalitatea din ocupație, dar putem impune reguli de validare să nu apară discrepanțe majore (ex: nu vrem ca un narator să spună că un șomer trăiește în lux sau că un profesor are salariu imens – aspecte ce țin de realism economic).

Graful de mai sus va ghida ordinea de eșantionare: vom începe cu variabile demografice macro (regiune, mediu, sex, vârstă, etnie), apoi educație, stare civilă, apoi ocupație, apoi nume, și în final generarea textelor narative pe baza acestora. Implementarea grafului se poate face fie **explicit, procedural** (cod care parcurge în ordine și folosește distribuții condiționate la fiecare pas), fie folosind

un model probabilistic integrat (PGM) care descrie dependendențele și permite eșantionare multivariată.

Generarea datelor structurate (PGM, IPF și tabele PxWeb utile)

Pentru generarea propriu-zisă a celor 100k profiluri structurate, vom folosi abordări de **modelare probabilistică** pe baza datelor recensământului:

- **Probabilistic Graphical Model (PGM)**: Așa cum au procedat NVIDIA și Gretel pentru dataset-ul din SUA, vom construi un model grafic (ex. o *rețea Bayesiană* sau *Markov*) care codifică relațiile din graful de mai sus [25]. De exemplu, putem crea noduri pentru fiecare variabilă (regiune, sex, vârstă, etc.) și tablice de probabilitate condiționată (CPT) pe baza statisticilor cunoscute. Unele CPT-uri pot fi extrase direct din tabelele recensământului:
- **Distribuția (Regiune × Urban/Rural)** – direct din date (știm total urban/rural pe regiuni). De ex., municipiul Chișinău este aproape 100% urban; regiunile Nord, Centru, Sud au combinații (vom lua datele din *"Populație rezidentă pe mediu și județ"*).
- **Distribuția (Vârstă × Sex)** – putem folosi piramida: BNS oferă tabel cu efectivul populației pe fiecare vârstă și sex [6]. Putem prelucra să obținem P(age=x | sex) sau direct P(age,sex).
- **Distribuția (Etnie × Regiune)** – disponibilă în caracteristicile etnoculturale (Ex.: BNS a dat ponderea etniilor pe regiuni de dezvoltare [20]). Putem construi un CPT P(etnie | regiune) conform acestor date (cu marje exacte: ex. dacă regiune=Nord, atunci probabilitate ~9,6% ucrainean, 3,5% rus etc [26]; dacă regiune=Găgăuzia, P(găgăuz) ~81,9%).
- **Distribuția (Educație × Vârstă × Sex × Urban)** – dacă datele permit. Posibil BNS are tabele cu nivel de studii absolvite pe grupe de vârstă, sex și mediu (cel puțin parțial, sau putem folosi IPUMS-like microdata dacă ar fi disponibil). Dacă nu, putem aproxima:
    - P(educație | age, sex, urban) = P(educație | age, urban) (presupunând sex influențează mai puțin, deși femeile au ușor educație mai înaltă [27]). Construim aceste distribuții pe intervale de vârstă (ex. 25-34, 35-44 etc. – de exemplu știm total 19,1% au studii sup. pe 10+ ani [28], dar la 25-34 ani poate fi ~30%, la 65+ doar ~10%). Ne putem folosi de comunicatul BNS: "fiecare a cincea persoană de peste 15 ani are studii superioare" (~20%) [29] și de faptul că în Chișinău e 36,7% cu studii superioare vs. 23,5% în Bălți [30]. Vom îngloba astfel: urban > rural la educație înaltă, generațiile tinere > vârstnici.
- **Distribuția (Stare civilă × Vârstă × Sex)** – datele pot fi culese din statistici demografice (ex. *Indicatori demografici, starea civilă pe vârste*). Recensământul 2014 a avut astfel de tabele; pentru 2024 BNS a comunicat deja agregatele [7]. Putem folosi IPF (vezi mai jos) pentru a combina margini:
    - margine P(marital) global (55.8% căsătoriți etc [7]),
    - distrib. pe vârstă a celor căsătoriți din date demografice (de ex. rata de căsătorie pe cohortă).

- **Distribuția (Ocupație × Educație × Vârstă × Sex × Regiune)** – aceasta e cea mai complexă. Vom simplifica printr-o abordare pe pași:

    1. Determinăm întâi *statutul de activitate*: angajat, șomer, inactiv (pensionar, elev, casnic). Acesta clar depinde de vârstă (pop. ocupată e în principal 20-60 ani) și de educație (șomerii mai mulți la educație joasă). Recensământul 2024 a oferit date economice: forța de muncă totală era X etc. Vom folosi rata de activitate (~40,2% din pop 15+ era activă, conform Telegraph.md [23]) pentru calibrări.
    2. Dacă activ = da, atunci alocăm o ocupație. Putem construi distribuții separate pentru categorii socio-profesionale: de ex. probabilitatea de a fi în agricultură, industrie, servicii condiționat de rural/urban și educație.

3. Ex.: un bărbat, 50 ani, rural, studii gimnaziale -> probabil agricultor sau muncitor calificat. O femeie 30 ani, urban, studii superioare -> probabil sector terțiar (profesoară, economistă, medic etc.).
4. Ne putem folosi de datele de ocupare din Anuarul Statistic: pondere agricol ~30% din ocupare totală, servicii ~50%, industrie ~20%, cu variații mare urban vs rural.
5. În interiorul fiecărui sector vom alege profesii concrete uniform sau după popularitate (ex.: în agricultură cele mai frecvente: fermier pe cont propriu, muncitor agricol; în servicii: vânzător, profesor, șofer, lucrător comercial, funcționar public etc.).
6. Dacă activ = nu și vârsta < 18 -> marcat elev; dacă 18-24 și continuă studiile -> student; dacă vârstă pensionare -> pensionar.
7. Astfel *ocupația* este ultimul câmp generat, integrând mai multe aspecte anterioare.

- **Iterative Proportional Fitting (IPF / raking)**: Vom aplica IPF pentru a aproxima distribuția comună a mai multor variabile când avem doar distribuții marginale și unele tabele parțiale. De exemplu:

- Avem margini P(regiune), P(urban), P(sex), P(age), P(etnie) individual, plus unele tabele bidimensionale P(regiune × etnie) [20], P(age × sex) etc. IPF ne poate construi o matrice multi-dimensională P(regiune × urban × sex × age × etnie) care să respecte toate aceste margini cunoscute (sau majoritatea). Vom rula IPF până la convergență, obținând o distribuție multi-dimensională din care putem eșantiona 100k instanțe (de fapt, IPF ne dă proporțiile exacte pentru fiecare combinație).
- Similar, pentru alt sub-set de variabile: P(age × educație × sex × urban) având margini P(educație) [10], P(age × urban) etc. sau P(age × marital × sex).
- IPF va fi util mai ales dacă decidem să **precalcificăm** un *micro-dataset sintetic complet* (ex. la nivel de individ) respectând toți indicatorii. Practic, am putea genera o micro-populație sintetică de 2,4 milioane de indivizi și apoi eșantiona 100k dintre aceia. Dar aceasta este costisitor; în schimb, putem aplica IPF doar pe eșantionul de 100k direct, iterativ: generăm inițial, calculăm distribuțiile obținute vs cele vizate și ajustăm selectiv eșantionul (prin resampling cu greutăți).
- Pentru eficiență, putem folosi un algoritm de *raking*: atribuim fiecărei înregistrări generate inițial o greutate și ajustăm greutățile până marginea ponderată bate cu totalurile țintă. Apoi eșantionăm din nou fără greutate, dar eliminând/înlocuind unele cazuri din categoriile supraponderate cu altele subponderate.

Surse PxWeb utile: Platforma PxWeb a BNS (statbank.statistica.md) conține tabele relevante: - **Populația rezidentă pe vârstă, sex și mediu, la 8.04.2024** – pentru distribuția de vârstă (posibil în *Population and demographic processes* → *Number of population*). - **Structura etnică a populației pe regiuni** – ar putea fi în *Census 2024* → *Caracteristici etnoculturale*. (Dacă nu, am folosit date din comunicat). - **Populația pe nivel de educație absolvit și vârstă** – în *Caracteristici educaționale*, dacă PxWeb are (dacă nu, datele din comunicat și anexe). - **Stare civilă pe grupe de vârstă și sex** – de verificat în *Caracteristici demografice*. - **Populația ocupată pe activități economice sau ocupații, pe medii** – eventual în *Caracteristici economice*. S-ar putea să existe tabele ca: distribuția populației ocupate pe sectoare (agric/ind/serv) pe județe sau mediu urban/rural. - **Dimensiuni medii ale gospodăriilor, număr de copii** – dacă vrem să generăm și informații familiale (nu este obligatoriu, dar putem verifica). Recensământul a colectat și număr de copii născuți de femei – ex. s-a comunicat că a crescut ponderea femeilor cu 2-3 copii [31]. Acest lucru poate fi integrat anecdotic în narațiuni (ex: „mamă a doi copii adulți” pentru o femeie de 60 ani, dacă se potrivește structurii).

În practică, vom extrage din PxWeb tabelele relevante (manual sau via API) și vom prelucra datele cu Python/R. De exemplu, pentru (age, sex): un tabel posibil este *POP107D – Population by age, sex, area* (ipoteză). Pentru (regiune, etnie): datele comunicate deja. Pentru educație, vom folosi eventual tabelul din anexa 4.1 (menționat în comunicatul educațional [32], disponibil ca Excel [33]).

Modelarea prin PGM/IPF ne asigură că **nu supra-eșantionăm** subgrupele mici. Fiecare combinație de caracteristici va apărea proporțional cu probabilitatea sa reală. De exemplu, persona de etnie romă vor fi ~0,4% [19] (deci ~400 din 100k), cele din Găgăuzia ~4,3% [2] (~4300 din 100k), etc. Acest lucru este esențial pentru reprezentativitate și evitarea introducerii de biais.

Șabloane de prompturi LLM pentru generarea narațiunilor (în română)

După ce avem profilul structurat al fiecărei persoane, vom folosi **șabloane de prompturi** în limba română pentru a ghida un **model de limbaj** (LLM) open-source să genereze textele narative (descrierile persona). Vom asigura template-uri specifice pentru fiecare segment narativ, astfel încât modelul să menționeze detaliile relevante din profil în mod coerent și natural. Iată planul de generare și exemple de prompturi:

1. **Pregătirea contextului pentru LLM:** Vom furniza modelului un rezumat al atributelor persoanei. Acesta poate fi sub forma unei scurte fișe (ex. „*Nume*: Ion Popescu; *Sex*: masculin; *Vârstă*: 45 de ani; *Etnie*: moldovean; *Religie*: ortodox; *Stare civilă*: căsătorit; *Educație*: studii superioare (inginerie); *Ocupație*: inginer constructor; *Localitate*: Chișinău, urban; *Limbă maternă*: româna; *Alte detalii*: pasionat de fotbal, doi copii, etc.”). Putem include și câteva trăsături de personalitate alese aleator (ex. „trăsături: disciplinat, sociabil, pragmatic”), pentru a fi inserate de model în text.
2. **Șablon general pentru generarea tuturor paragrafelor o dată:** O abordare ar fi să cerem modelului să genereze întreg profilul narativ format din multiple paragrafe tematice, pentru a garanta coerența între ele. Exemplu de prompt (pseudo-limbaj, în română):

```
Ai următoarele informații despre o persoană fictivă:
- Nume: {name}
- Sex: {sex}
- Vârsta: {age} ani
- Ocupație: {occupation}
- Educație: {education_level}
- Stare civilă: {marital_status}
- Localitate: {city}, regiunea {region} ({residence_type})
- Etnie și limbă maternă: {ethnicity}, limba maternă {mother_tongue}
- Religie: {religion}
- Trăsături de personalitate: {trait1}, {trait2}, {trait3}
- Hobby-uri: {hobby1}, {hobby2}, {hobby3}
Redactează un profil narativ coerent al acestei persoane, compus din 6
paragrafe:
1. **Descriere generală** – Prezintă pe scurt cine este {name},
incluzând vârsta, ocupația și trăsături de personalitate esențiale.
2. **Profil profesional** – Detaliază viața profesională: locul de
muncă, abilități, responsabilități, etica în muncă și aspirații de
carieră ale persoanei.
3. **Hobby-uri sportive și stil de viață activ** – Descrie ce sporturi
sau activități fizice practică {name}, cum își petrece timpul liber
activ și eventual echipe preferate sau rutine de sănătate.
4. **Interese culturale și artistice** – Descrie gusturile în materie de
cultură: muzică, arte, lectură, filme sau alte pasiuni creative ale
persoanei.
```

```
5. **Obiceiuri de călătorie și vacanțe** – Menționează dacă îi place să călătorească, ce fel de destinații sau experiențe de călătorie preferă, eventual un vis de călătorie.
6. **Obiceiuri culinare și tradiții** – Prezintă relația persoanei cu gastronomia: dacă gătește, ce fel de bucătărie preferă (tradițională, internațională), mâncăruri sau băuturi favorite, eventual cum aceste obiceiuri se leagă de cultura sa.

Folosește un ton realist și detalii relevante care se potrivesc profilului (de ex., menționează elemente legate de regiunea de proveniență sau cultura {ethnicity}). Nu repeta informațiile mot-à-mot, ci integrează-le natural. Fiecare paragraf trebuie să înceapă cu numele persoanei sau un pronume și să aibă 2-5 fraze bine formulate în limba română.
```

Acest prompt detaliat îi spune modelului exact ce secțiuni să scrie și ce informații să includă. Modelul va produce un text cu 6 paragrafe separate, conform cerinței.

Avantajul generării tuturor paragrafelor într-un singur prompt este consistența – modelul poate face referințe între aspecte (ex. hobby-urile se pot lega de background cultural, planurile de carieră pot apărea în secțiunea profesională și la cea de călătorii sub forma „economisește bani pentru...", așa cum am văzut în exemplul real [34]).

1. **Șabloane individuale pentru fiecare paragraf (alternativ):** Dacă modelul are tendința să producă un singur bloc text neîmpărțit, putem genera secvențial, paragraf cu paragraf. În acest caz, după fiecare generare, includem textul anterior ca context, ca modelul să nu se contrazică. Exemple:
2. *Descriere generală:*

```
Informații profil: {sex}, {age} ani, {occupation}, {marital_status}, {city}/{region}, trăsături: {trait1}, {trait2}.
Scrie o descriere succintă la persoana a treia despre această persoană, incluzând vârsta, ocupația și câteva trăsături de caracter.
```

3. *Profil profesional:*

```
{name} lucrează ca {occupation}. Continuă povestea, descriind abilitățile sale profesionale, stilul de lucru și aspirațiile în carieră într-un paragraf.
```

4. *Hobby sport:*

```
Descrie în 2-3 propoziții ce activități sportive sau recreative face {name} în timpul liber, având în vedere vârsta și contextul său.
```

5. *Interese culturale:*

```
Ce fel de muzică, artă sau literatură preferă {name}? Scrie un paragraf
despre hobby-urile culturale ale persoanei (muzică, filme, cărți etc.),
legându-le de personalitatea și originea sa.
```

6. *Călătorii:*

```
Scrie 2-3 fraze despre preferințele lui/ei {name} în materie de
călătorii: unde îi place să meargă, cu cine, și cum planifică vacanțele.
```

7. *Culinar:*

```
Descrie relația lui/ei {name} cu gastronomia: știe să gătească? ce
feluri de mâncare preferă? menționează eventual preparate tradiționale
din zona/regiunea sa.
```

8. *Fundal cultural:*

```
Menționează în 2 propoziții originile familiale și cultura în care a
crescut {name} (de ex. limba vorbită acasă, obiceiuri religioase, mediul
- sat/oras - și valorile transmise).
```

Vom scrie prompturile în română, deoarece dorim ca ieșirea să fie în română și un model multilingual (ex. bazat pe Llama 2 sau Mistral) va înțelege. Putem include variabilele în limba română direct (ex. „sex: masculin” etc.) pentru claritate.

În practică, vom folosi un model de limbaj open-source, cum ar fi **Llama 2** sau **Mistral 7B/13B** fine-tuned pe instruct, ori alte modele cu abilități solide în limba română. (Dacă e necesar, putem traduce intern promptul din română în engleză pentru un model puternic englez și apoi traduce rezultatul înapoi – însă există modele suficient de bune și direct în română).

Exemple de **ieșiri așteptate** (scurte, ilustrative): - *Descriere generală:* „**Maria Ionescu** are 34 de ani și este medic de familie într-o comună din regiunea de Centru. Este o fire empatică și dedicată, respectată de comunitate pentru modul în care îmbină profesionalismul cu compasiunea. Maria este totodată foarte organizată și perseverentă, calități pe care și le reflectă atât în carieră, cât și în viața de zi cu zi.” - *Profil profesional:* „În rolul său de medic, **Maria** dă dovadă de un devotament remarcabil. Și-a dezvoltat abilități solide de diagnostic și comunicare cu pacienții, având o răbdare și o atenție la detalii ieșite din comun. Munca în mediul rural vine cu provocări, însă Maria colaborează strâns cu asistenții medicali comunitari și organizează periodic mici campanii de informare privind sănătatea. Ambiția ei este să aducă servicii medicale de calitate cât mai aproape de oamenii din sat și, pe termen lung, să creeze un centru medical modern în zonă.” - *Hobby sport:* „**Maria** acordă o mare importanță sănătății și formei fizice. În fiecare dimineață face o plimbare rapidă pe ulițele satului, iar în weekend participă la un grup informal de alergare prin pădurea din apropiere. De asemenea, joacă volei ocazional cu colegii de la dispensar, considerând mișcarea o metodă excelentă de a-și limpezi mintea după o săptămână încărcată.” *(și așa mai departe pentru celelalte segmente).*

Notă: Vom ajusta prompturile și parametrii modelului (ex. *temperature*, *max_tokens*) pentru a obține consistență și diversitate. Vom verifica dacă modelul tinde să repete structuri sau fraze – dacă da,

putem adăuga variație în prompt (de exemplu, oferind și un exemplu de output așteptat). Un alt truc este utilizarea unor *few-shot examples*: îi putem arăta modelului un exemplu scurt (input + output) al unui profil fictiv generat corect, pentru a-i seta stilul (similar tonului din exemplele Nemotron, care sunt detaliate și pozitive).

Logică de validare: coerență narativă și acuratețe statistică

După generarea tuturor câmpurilor (structurate și text), vom aplica un modul de **validare** pentru a ne asigura că fiecare profil sintetic este atât **coerent intern**, cât și că ansamblul de 100k respectă proprietățile statistice globale.

Validare conținut narativ vs date structurate: - **Consistență identitate:** Numele persoanei și pronumele de gen trebuie să fie corecte peste tot. De exemplu, dacă `sex=feminin` și prenume „Maria”, textul să nu o menționeze ca „el”. Vom implementa o verificare automată (un script care caută incongruențe, eventual folosind o abordare NLP simplă: pronume nepotrivite). - **Concordanță vârstă – poveste:** Vom valida că narațiunea se potrivește vârstei. Exemple de erori de evitat: - Un copil de 5 ani nu poate avea „10 ani experiență ca inginer”. Dacă profilul are `age < 14`, narațiunea nu trebuie să conțină secțiuni de carieră sau detalii nepotrivite. Soluție: pentru minorii <18, *șabloanele de prompt* vor fi adaptate – de ex., paragraful „profil profesional” devine „profil educațional” (descriind școala, materiile preferate, vise de viitor), paragraful de hobby-uri accentuează activități de copil etc. Vom implementa o ramificare: dacă age<18, atunci generează narațiuni special pentru un personaj copil/adolescent (fără job, cu părinți/școală menționate). - Vârstnici: dacă `age` este, să zicem, 80, textul ar trebui să indice pensionare, nepoți, amintiri, nu că respectiva persoană „conduce departamentul de marketing” (foarte puțin probabil). Vom adapta promptul: la `age>65`, secțiunea profesională poate vorbi la trecut („a lucrat ca… înainte de pensie și este respectat pentru…”), iar secțiunea de hobby poate menționa activități potrivite (grădinărit, întâlniri cu familia). Un modul de validare va scana outputul vârstnicilor după cuvinte-cheie (ex „pensionar”, „nepoți”) și dacă lipsesc, putem revizui manual câteva prompturi. - **Educație vs ocupație vs text:** Vom verifica corelația dintre nivelul de studii și ocupația descrisă în text. De exemplu, dacă `education_level = gimnazial` dar modelul a creat un paragraf profesional în care persoana e „doctor în fizică” – e o inadvertență. Astfel de cazuri ar fi rare dacă promptul include educația explicit, dar pentru siguranță vom face un pass de validare: - Folosind un dicționar de ocupații ce necesită studii superioare (ex: doctor, inginer, avocat etc.) – dacă apare o astfel de ocupație în text, dar `education_level` al persoanei nu este universitar, vom marca profilul pentru corecție (fie îi ridicăm `education_level`, fie ajustăm ocupația). - Invers, dacă cineva are `education_level = universitar` dar textul spune „muncește ca zilier la fermă” – e neobișnuit dar nu imposibil. Totuși, vom semnala pentru eventuală modificare (putem fie schimba ocupația la ceva potrivit studiilor, fie lăsa ca excepție – un absolvent care lucrează sub calificare nu e imposibil, deci nu vom forța eliminarea, doar ne asigurăm că nu e un tipar frecvent). - **Stare civilă vs text:** Dacă `marital_status = căsătorit(ă)`, textul ar trebui să menționeze sau să fie compatibil (ex. să nu spună „își caută sufletul pereche”). Nu e necesar ca fiecare profil să menționeze explicit soțul/soția, dar dacă o face, să se potrivească. Vom scana după indicii: cuvinte precum „soț/soție, copii” comparativ cu starea civilă. Dacă cineva e `necăsătorit` dar textul zice „soțul ei…”, clar e o eroare. - De asemenea, `divorțat` vs text – dacă modelul a omis mențiunea, nu e neapărat problemă, dar ar fi frumos să reflecte în personalitate („după un divorț dificil, și-a crescut singur copiii…” etc.). Putem îmbunătăți promptul să includă pentru divorțați/văduvi o notă: „menționează pe scurt situația (ex: văduv – a pierdut soția…)”. - **Copii vs vârstă/ stare civilă:** Dacă în text apar „doi copii mici”, ne asigurăm că persoana are vârsta compatibilă (ex: >= 20 ani, realist >=25). Dacă apare „nepoți”, persoana ar trebui să aibă probabil 50+ ani. Vom scrie mici reguli pentru aceste aspecte (ex: dacă găsim cuvântul „bunică/bunic” în text, age < 40 -> semnalăm). - **Consistența listelor de hobby/skill cu textul:** Vom genera listele *post factum* din text, ideal automat. De exemplu, putem rula o rutină care identifică substantivele cheie din paragraf de hobby și le pune în

listă. Vom valida că fiecare element din `skills_and_expertise_list` apare în textul profilului profesional (și invers, dacă textul menționează abilități distincte, lista ar trebui să le reflecte pe toate). Similar pentru hobby-uri. Aceasta se poate face cu un parser simplu sau chiar cu un LLM specializat pe extragere. Vom itera până ce listelor corespund conținutului (ex.: dacă lipsesc unele hobby-uri din listă, le adăugăm). - **Limbaj și ton:** Vom inspecta manual un eșantion de output-uri pentru calitatea limbii române (coerență, diacritice, stil). Modelul ales trebuie să poată produce text cursiv cu diacritice. Dacă nu, vom adăuga un post-procesor care să corecteze diacriticele (există unelte pentru asta). De asemenea, ne vom asigura că niciun text nu conține informații jignitoare sau senzitive nepotrivite. Fiind generație sintetică instruită, riscul e mic, dar vom filtra eventual: nu vrem biasuri negative față de un grup (ex.: să nu cumva modelul să producă stereotipuri ofensatoare despre romi sau altele). Prompturile bine calibrate ar trebui să evite asta, dar vom efectua un scanning (poate chiar cu un model de clasificare de toxicitate în română) și ajusta/re-genera acolo unde e cazul. **Licența finală CC BY** ne obligă să fim atenți la conținut problematic pentru uz public.

Validare distribuțională globală: - După generarea întregului set de 100.000 persoane, vom calcula distribuțiile finale ale atributelor și le vom compara cu cele țintă: - Proporția de femei vs bărbați ar trebui să fie ~52.8% vs 47.2% [4] . Dacă observăm 53.1% vs 46.9%, e acceptabil (diferență minoră), dar dacă e 56% vs 44%, atunci clar trebuie ajustare. Vom re-echilibra selectând aleator câteva profiluri de sex majoritar și schimbându-le sexul (și prenumele + textul asociat) sau regenerând până se atinge ținta. De preferat însă, design-ul de eșantionare va preveni deviații mari. - Distribuția pe vârste: Vom grupa vârstele generate și verifica vs piramida reală. Vom aplica un test chi-pătrat sau Kolmogorov-Smirnov între distribuția sintetică și cea reală pe grupele 0-14, 15-64, 65+ (și poate sub-grupe de 5 ani). Ne așteptăm la diferențe minore date de întâmplare. Dacă vreun segment e semnificativ off (ex: avem 20% copii în loc de 19.2% – probabil acceptabil; dar dacă e 25%, trebuie resampling). Putem remedia prin *raking*: atribuim fiecărui profil un factor de ajustare și resampling până când cifrele se aliniază mai bine. - Distribuția etniilor și limbilor: Vom număra câți moldoveni/ruși etc. avem și verifica vs procentul țintă [19] . Ar trebui să corespundă aproape exact, deoarece intenționat le generăm proporțional. Dacă mici diferențe, putem re-eticheta câteva persoane borderline (ex: unii declarați „român” vs „moldovean” se pot ajusta deoarece oricum cultural sunt similari). Scopul e să nu avem supra-reprezentare a vreunui grup mic – ex: fix ~4% găgăuzi. Similar pentru religii – deși dacă nu le folosim direct în profil decât implicit, nu e crucial la virgulă, dar vom verifica că ~95% profiluri cel puțin nu contrazic – adică majoritatea să reiasă implicit ortodoxe (prin context cultural). - Distribuția educației: Vom compara cu datele din comunicate (19,1% studii superioare etc. [28]). Aici pot apărea mici abateri, mai ales dacă modelul LLM a plasat mai multe personaje cu studii înalte decât trebuia (LLM ar putea fi tentat spre „pozitiv”, dând multora studii superioare). De aceea, e critic să eșantionăm educația înainte și să o blocăm ca atare. Vom corecta manual dacă vedem deviații (ex: dacă rezultă 25% cu studii superioare în set, vom reconverti o parte la studii medii, eventual regenera narațiunile respective). - Distribuția ocupațiilor: Aceasta e dificil de validat direct vs real, dat fiind avem sute de ocupații. Dar putem valida la nivel de *sectoare* sau *clase*: de ex. număr de agricultori vs număr de IT-iști. Ne așteptăm ca <2% să fie programatori în set (câteva sute, reflectând realitatea micii comunități IT în populatie generală), iar agricol să fie destui (să zicem 15% dacă includem și fermieri de subzistență). Dacă constatăm că LLM-ul a generat ocupații „fancy” prea des, vom reinstrui partea de ocupație. Totodată, verificăm **frecvența fiecărei ocupații**: nu vrem dubluri excesive. De ex., să nu avem 500 de profile cu același job *„profesor de matematică”* cu aceleași fraze – ceea ce ar indica lipsă de diversitate. Vom agrega pe `occupation` și dacă top 10 ocupații acoperă peste, să zicem, 20% din set, e suspicios (populația reală e diversă, primele 10 ocupații poate ating 10-15% din total). Dacă identifcăm modele repetitive, putem interveni diversificând prompturile sau folosind *temperature* mai mare la LLM pentru mai multă variație lexicală. - **Diverse:** Vom asigura că nu am pierdut proporția urban/rural – ar trebui să iasă ~46/54%. Verificăm distribuția regională – exact 5 categorii cu procentele date [2] ; dacă ușor off, aplicăm corecții minore (ex: înlocuim câteva profile din regiunea suprapopulată cu altele din cea subțipopulată, eventual re-generând localitatea și adaptând un pic textul acelor persoane să se potrivească noii regiuni).

Validare consistență tehnică: - Formatul final Parquet va fi verificat: toate rândurile au același număr de câmpuri, tipurile de date corecte (ex: `age` int64, cele narative string). Vom verifica că encoding-ul e UTF-8 pentru text (diacriticele în română). - Fiecare profil are UUID unic – vom rula un check de unicity pe coloana `uuid`. - Vom verifica absența datelor lipsă neintenționate: ex. nu vrem vreun `occupation` gol (în afară de cazuri deliberate ca elev – dar atunci punem explicit „Elev"). - **Coerența între câmpuri structurate redundant**: ex: dacă avem și `region` și `district`, să corespundă (facem un cross-check că fiecare district aparține region-ului indicat). Dacă găsim nepotriviri (posibile dacă am schimbat regiunea unor profile fără să schimbăm localitatea), le corectăm. - Aplicație practică: putem scrie un script de validare unitară (unit tests) care să parcurgă datasetul și să facă aceste asercții și rapoarte de eroare.

Packaging final: Parquet, metadata, licență, documentație

După generare și validare, urmează **ambalarea dataset-ului** într-o formă utilă pentru distribuire și reutilizare:

- **Format Parquet:** Vom salva datele în format Parquet, partajat eventual în bucăți (dacă e foarte mare, dar 100k nu e enorm – se poate un singur fișier). Parquet asigură eficiență la citire și tipizare clară a coloanelor. Vom folosi librăria *pandas* sau *pyarrow* în Python pentru a crea tabelul. Câmpurile text (narative) vor fi stocate ca `BYTE_ARRAY` (string) cu encoding UTF8. Vom comprima Parquet-ul (snappy compression implicit).
- **Metadate și documentație:** Vom pregăti un fișier README (Markdown) și eventual un *data card* (dacă găzduim pe HuggingFace). Acesta va descrie:
- Scopul dataset-ului (antrenare AI, simulare de utilizatori diverși).
- Modul de generare (scurt: bazat pe recensământ 2024, sintetizat cu PGM și LLM, fără date personale reale).
- Structura câmpurilor (vom lista fiecare coloană cu semnificație, eventual un dicționar de categorii unde e cazul).
- Exemplu de utilizare (cum se citește Parquet-ul și se accesează un profil).
- Informații despre calitate și limitări cunoscute (ex: „persoanele sunt fictive, orice asemănare e coincidență; distribuția este calibrată la nivel macro – pot exista abateri minore la nivel micro față de realitate").
- **Licența** dataset-ului: se va opta pentru **CC BY 4.0** (Creative Commons cu atribuire) – la fel ca dataset-ul Nemotron-Personas [15], ceea ce permite utilizare comercială largă, cu condiția atribuirii sursei. Vom include nota de licență în documentație.
- Surse: vom menționa BNS ca sursă a datelor statistice și eventual NVIDIA Nemotron ca inspirație metodologică.

- Informații despre versiuni și update-uri: dataset-ul e extensibil – dacă în viitor apar date 2030 sau decidem extinderea la alte țări, putem descrie cum se va face.

- **Scalabilitate:** Deși acum generăm 100k profile, pipeline-ul este conceput să fie scalabil. Putem crește ușor la 1 milion sau mai mult, având grijă la puterea de calcul:

- Generarea datelor structurate și eșantionarea se face rapid chiar și pentru milioane, întrucât PGM/IPF sunt eficiente (eventual folosind numpy/pandas sau C/C++ backend).
- Bottleneck-ul este generarea cu LLM a narațiunilor, care pentru 100k persoane poate fi costisitoare. Pentru a scala, vom:
    - Folosi *batch inference* dacă modelul permite (de ex., generarea textului în paralel pe GPU, sau folosirea mai multor GPU-uri).

    - Opta pentru un model mai mic dar optimizat (ex: un fine-tune pe dedidcat 7B parametri care să fie foarte bun la aceste profiluri, rulând la ~0.5 sec per profil – deci 100k în câteva ore pe un server cu 4 GPU).
    - Putem genera *mai întâi un număr mai mic de profile* (ex 50k), evalua diversitatea textelor, apoi antrena un model de limbaj mai mic pe acele exemple (teacher-student) ca să genereze restul mai repede. Acesta e un truc avansat (folosit și în Nemotron probabil pentru volum mare).

- Orchestrat cu instrumente open-source: scriere de cod Python pentru flux, posibil integrare cu *Luigi/Airflow* pentru pipeline; paralelizare cu *multiprocessing* sau *Dask* pentru a genera în paralel sub-liste de persoane.

- **Verificare finală și exemplu de utilizare:** Vom încărca fișierul Parquet rezultat într-un mediu de test și vom extrage aleator câteva profiluri, atât pentru a inspecta manual (să vedem calitatea textelor), cât și pentru a demonstra în documentație cum arată. De exemplu, vom afișa un rând în format JSON:


```
{
  "uuid": "123e4567-e89b-12d3-a456-426614174000",
  "name": "Ion Ciobanu",
  "sex": "Masculin",
  "age": 52,
  "marital_status": "căsătorit",
  "education_level": "profesional (școală de meserii)",
  "occupation": "șofer de autobuz",
  "city": "Ungheni",
  "region": "Centru",
  "residence_type": "Urban",
  "cultural_background": "Ion a crescut într-o familie de moldoveni din Ungheni, într-un cartier muncitoresc, unde valorile precum hărnicia și respectul față de tradiție erau la loc de cinste.",
  "descriere_generala": "Ion Ciobanu, în vârstă de 52 de ani, este un bărbat calm și pragmatic, cunoscut în comunitatea sa drept un om pe care te poți baza. Lucrează ca șofer de autobuz în Ungheni și își iubește meseria pentru că îi permite să interacționeze zilnic cu oamenii din oraș.",
  "profil_profesional": "În cariera sa de peste 30 de ani ca șofer profesionist, Ion a parcurs sute de mii de kilometri fără incidente majore. Stăpânește foarte bine regulile de circulație și are reflexe excelente, câștigând respectul colegilor pentru punctualitate și seriozitate. Este mândru că a putut ajuta generații de elevi să ajungă zilnic la școală în siguranță. Visul lui Ion este ca într-o zi să contribuie la instruirea tinerilor șoferi, împărtășind din experiența sa bogată.",
  "hobby_sport": "Deși petrece multe ore pe drum, Ion se menține activ. În weekend joacă fotbal cu prietenii din copilărie și este un microbist pasionat, urmărind meciurile echipei naționale împreună cu fiul său. De asemenea, îi place să facă drumeții scurte pe dealurile din jurul orașului, bucurându-se de natură.",
  "hobby_arta_cultura": "Ion are o latură artistică neașteptată: îi place muzica populară moldovenească și nu ratează niciun festival
```

```
folcloric din raion. Uneori, serile și le petrece citind cărți istorice
despre regiunea Moldovei, fiind fascinat de poveștile locale. Împreună
cu soția sa, merge la casa de cultură când sunt spectacole de teatru sau
cinema ambulant în oraș.",
  "hobby_calatorii": "Pentru Ion, cele mai frumoase călătorii sunt cele
în sânul naturii de acasă. Vara își vizitează rudele de la țară, unde îl
așteaptă cu un peisaj pitoresc pe malul Prutului. Totuși, are și un vis
mai îndrăzneț: să ajungă într-o zi să vadă Marea Neagră la Odessa, loc
despre care a auzit multe povești în tinerețe.",
  "hobby_culinar": "Fiind născut într-o familie de bucătari pricepuți,
Ion a prins gustul bucatelor tradiționale. Gătește cu plăcere zeamă și
sarmale împreună cu soția, mai ales de sărbători când casa se umple de
copii și nepoți. În plus, este pasionat de vinurile de casă; are o mică
vie la marginea orașului și în fiecare toamnă produce câteva damigene de
vin roșu, mândru de rodul pământului și al muncii sale.",
  "skills_and_expertise_list": ["condus defensiv", "punctualitate",
"cunoașterea traseelor locale", "mentenanța vehiculului",
"comunicare cu publicul"],
  "hobbies_and_interests_list": ["fotbal", "drumeții", "muzică
populară", "lectură istorică", "viticultură artizanală"]
}
```


Acesta este un exemplu ipotetic de ieșire JSON (valorile sunt generate conform regulilor). Observăm coerența: sex, vârstă, ocupație se aliniază cu povestea; hobby-urile din listă sunt vizibile în texte. Astfel de exemple vor fi incluse în documentație pentru ca utilizatorii să înțeleagă structura.

- **Testare de utilizare**: Vom include instrucțiuni cum poate fi folosit dataset-ul, de exemplu:

- *"Puteți filtra personas după câmpurile structurate. Ex: extrageți toți indivizii de sex feminin, știind că reprezintă ~52,8%* [4] *din eșantion, și antrena un model pe narațiunile lor. De asemenea, puteți utiliza câmpurile* `skills_and_expertise_list` *și* `hobbies_and_interests_list` *pentru a crea etichete multi-clasă în aplicații de recommender sau pentru a verifica că un asistent virtual oferă răspunsuri personalizate pe interese. Narațiunile pot fi folosite direct la* prompting *pentru a simula conversații cu utilizatori fictivi din diverse medii."*

În concluzie, pipeline-ul propus folosește date **oficiale recente** (Recensământ 2024) pentru a ancora demografia, aplică metode solide (PGM/IPF) pentru a genera *100.000 de profiluri sintetice* **realist distribuite**, și utilizează puterea modelelor de limbaj deschise pentru a crea descrieri bogate în limba română. Vom obține astfel un dataset valoros, comparabil ca utilitate cu Nemotron-Personas, dar personalizat pe Moldova, care poate fi extins pe viitor și către alte țări sau dimensiuni (domenii specifice sau evoluție în timp), contribuind la dezvoltarea AI cu date locale sintetice **de înaltă fidelitate** [1] [25].

Surse utilizate:

- Date demografice și distribuții reale din rezultatele finale ale Recensământului 2024 (BNS): structura pe sex și mediu [4], distribuția pe regiuni [2], structura etnică națională și pe regiuni [19] [20], grupelor de vârstă și starea civilă [7] [6], niveluri educaționale [8] [10], religie [22] etc.
- Prezentarea dataset-ului *Nemotron-Personas* (NVIDIA) ca referință de concept și structură [3] [25]. Acesta confirmă utilizarea combinată a PGM și LLM pentru generarea de personas sintetice și include exemple de câmpuri narative folosite ca model.

- Alte informații contextuale din comunicate de presă și analize (UNFPA, EU Neighbours) privind recensământul [2] și evoluții demografice (educație, migrație) pentru a calibra mai bine relațiile dintre variabile.

Acest design de pipeline este fezabil cu unelte open-source disponibile în 2025-2026, necesitând doar date publice și modele open (ex. *pandas/pyarrow* pentru date, *pgmpy* sau *numpy* pentru PGM/IPF, *HuggingFace Transformers* pentru LLM, etc.). Astfel, putem genera setul de **100k personas** și chiar extinde la milioane la nevoie, totul într-un mod transparent și reproductibil, cu documentație adecvată și licență liberă (CC BY 4.0) [15] pentru a maximiza reutilizarea de către comunitatea AI și factorii de decizie.

---

[1] [3] [15] [25] Nemotron-Personas: Improve AI Training With the First Synthetic Personas Dataset Aligned to Real-World Distributions
https://huggingface.co/blog/nvidia/nemotron-personas

[2] Over 2.4 million residents in the Republic of Moldova, of whom 52.8% are women – First final results of the 2024 Population and Housing Census | United Nations in Moldova
https://moldova.un.org/en/299360-over-24-million-residents-republic-moldova-whom-528-are-women-%E2%80%93-first-final-results-2024

[4] [19] [20] [21] [22] [26] Recensământul populației din 2024 (Republica Moldova) - Wikipedia
https://ro.wikipedia.org/wiki/Recens%C4%83m%C3%A2ntul_popula%C8%9Biei_din_2024_(Republica_Moldova)

[5] [6] [7] [31] Final results of the 2024 Population and Housing Census: Demographic Characteristics of the Population
https://statistica.gov.md/en/final-results-of-the-2024-population-and-housing-census-demographic-characterist-10121_61953.html

[8] [9] [10] [24] [28] [32] [33] Rezultatele finale ale Recensământului Populației și Locuințelor 2024: Caracteristici educaționale ale populației
https://statistica.gov.md/ro/rezultatele-finale-ale-recensamantului-populatiei-si-locuintelor-20241-caracteri-10121_61952.html

[11] [12] [14] [16] [17] [18] [34] nvidia/Nemotron-Personas-USA · Datasets at Hugging Face
https://huggingface.co/datasets/nvidia/Nemotron-Personas-USA

[13] Final results of the 2024 Population and Housing Census: Geographical Distribution of the Population
https://statistica.gov.md/en/final-results-of-the-2024-population-and-housing-census-geographical-distributio-10121_61877.html

[23] [27] Recensământul 2024: Doar 40,2% din populația de peste 15 ani ...
https://telegraph.md/recensamantul-2024-doar-402-din-populatia-de-peste-15-ani-este-angajata-in-campul-muncii/

[29] Recensământ 2024: 7.500 de persoane din R. Moldova sunt ...
https://moldova1.md/p/55813/recensamant-2024-7-500-de-persoane-din-r-moldova-sunt-analfabete

[30] 99,6 la sută; Tot mai mulți moldoveni au studii superioare - Jurnal.md
https://www.jurnal.md/ro/news/a143a84189551829/rata-de-alfabetizare-in-r-moldova-99-6-la-suta-tot-mai-multi-moldoveni-au-studii-superioare.html