PEPM - Projet Étude Politique Master (M1)

UE : M1SOL023 - Méthodologie de la Recherche (Langue & Informatique, 2025-2026) Projet : Classification et Modèles Multimodaux Auteur : Adrien VERGNE & Yi Fan Version pipeline : 5.9.6 (CPU) Collecte des données : septembre 2025 Machine de référence : AMD Ryzen 7 7840U (16 threads), 96 Go RAM, exécutions CPU Date : 10 janvier 2026 **Dépôt GitHub : https://github.com/TeaS0710/PEPM_M1srbn-project-PEMP_V5.9.6-CPU-main/tree/main**

Table des matières

1. Introduction

1.1 Contexte et motivation

Les travaux en classification politique et en analyse de discours montrent qu'il est possible d'obtenir de bonnes performances à partir de textes (articles, communiqués, discours). Un enjeu majeur est toutefois la **validité de l'inférence** : de bons scores ne prouvent pas, à eux seuls, que le modèle apprend une « sémantique politique » au sens strict. Il peut capturer un **signal composite** fait de signaux corrélés à l'orientation (style rédactionnel, provenance/source, noms propres, thèmes récurrents, registre oral/écrit, conventions de mise en forme, etc.). Le projet PEPM se situe précisément à ce point de tension : obtenir des résultats reproductibles sur des données originales, tout en instrumentant le pipeline pour caractériser ce que le modèle apprend réellement.

1.2 Idée initiale (objectif scientifique visé)

L'idée initiale de PEPM était d'étudier la polarité et les nuances idéologiques **via la sémantique** :

- au niveau **lexical** (champs sémantiques, cooccurrences, marqueurs),
- au niveau **phrastique** (structures, patterns rhétoriques),
- et au niveau **acteur/source** (comparaison de sites et entités politiques), avec l'objectif de distinguer droite/gauche puis, à terme, des catégories plus fines (extrêmes).

1.3 Évolution du projet : d'une étude sémantique à un outil expérimental reproductible

À partir de la collecte (septembre 2025) et de la construction de corpus volumineux, le projet a évolué sous la contrainte du **temps de calcul**, de la **gestion matérielle** (RAM/CPU), et de la nécessité de **reproductibilité**. La priorité est devenue la construction d'un pipeline robuste et paramétrable permettant :

1) de transformer des sources hétérogènes en données d'apprentissage cohérentes (TEI -> TSV), 2) de lancer des expériences comparables via une **commande universelle** et des profils YAML, 3) d'explorer plusieurs modèles / paramètres sans casser l'ensemble, 4) de produire des artefacts d'évaluation (métriques, matrices de confusion, audits anti-fuite).

Compromis assumé : certaines parties « supérieures » (analyses plus riches, automatisations de figures avancées) ne sont pas allées jusqu'au bout par manque de temps, mais le cœur (core) est stable, générique et orienté reproductibilité.

1.4 Problématique (alignée sur l'état réel du système)

Problématique retenue :

> Dans quelle mesure une chaîne complète de classification (préparation instrumentée -> représentations -> apprentissage -> évaluation), appliquée à des corpus web et à de l'oral transcrit (ASR), permet-elle de distinguer

une orientation politique globale et de caractériser la nature du signal appris (signal composite : lexique, style, provenance, registre), notamment en présence de changement de domaine ?

1.5 Contributions

- **Données originales** : scraping web massif + sous-corpus ASR (YouTube -> Whisper) ; archives riches favorisant la reproductibilité.
- **Pipeline scalable et user-friendly** : profils YAML + commande universelle, `dataset_id` traçable, séparation prepare/train/evaluate.
- **Exploration de méthodes** : baselines fortes (TF-IDF + SVM/SMO/perceptron/arbres) + réseau léger spaCy ; Transformers envisagés mais limités par l'exécution CPU.
- **Blindage méthodologique** : audits de splits (anti-fuite), déduplication configurable, métriques et artefacts systématiques.

1.6 Périmètre des expériences du rendu

Le rendu se concentre sur :

- **Ideology_global (binaire gauche/droite)** : baseline principale.
- **Crawl (multi-classes ~20 sources/crawls)** : tâche diagnostique pour quantifier l'empreinte « source/ligne éditoriale ».
- **Mix web+ASR** : mesure de robustesse face au changement de modalité.

2. État de l'art

Cette section sert à situer le projet par rapport aux baselines classiques, aux approches embeddings/LLM, et aux difficultés de domain shift et de confondants.

2.1 Classification politique et biais médiatique

Les travaux en classification politique montrent qu'il est possible d'obtenir de bonnes performances avec des approches simples (sac de mots, TF-IDF, classifieurs linéaires). Cependant, la littérature insiste sur le fait que la performance peut provenir de signaux indirects : vocabulaire propre à une source, noms d'acteurs, cadrage thématique, ou style. D'où l'importance de protocoles d'évaluation robustes (splits, tests hors domaine, analyse des erreurs).

2.2 Représentations : sac de mots vs embeddings (LLM)

- **Sac de mots / TF-IDF** : interprétables, efficaces, rapides sur CPU ; souvent une baseline difficile à battre.
- **Embeddings / Transformers (type BERT/CamemBERT)** : potentiellement meilleurs en généralisation sémantique, mais coût computationnel élevé, nécessité de maîtriser la troncature et la longueur des séquences.

2.3 Multimodal et domain shift (oral transcrit)

L'oral transcrit (ASR) introduit du bruit et une distribution linguistique différente (registre, segmentation). On observe typiquement une baisse de performance par rapport à l'écrit, ce qui justifie des métriques par sous-corpus et des évaluations équilibrées.

2.4 État de l'art ciblé et ancrage méthodologique (sélection)

Cette section ne vise pas l'exhaustivité bibliographique, mais un cadrage *utile* pour justifier : (i) la construction d'un **corpus original** par scraping/normalisation, (ii) l'intérêt d'un **pipeline reproductible** (contrôles, anti-fuite, profils), (iii) et l'angle analytique réel du projet : une classification politique **multi-factorielle** (sémantique + cadrage + style + signatures éditoriales), plutôt qu'une "sémantique pure".

2.4.1 Mesurer l'idéologie et le positionnement politique à partir de textes

La littérature en science politique computationnelle montre que des régularités lexicales, rhétoriques et thématiques permettent d'estimer un positionnement (partis, médias, acteurs). Avant les Transformers, cette idée s'est cristallisée autour de méthodes de *scaling* (Wordscore, Wordfish) et d'analyses "words-as-data", encore utiles pour discuter **interprétabilité** et **hypothèses**.

2.4.2 Détection de biais médiatique / hyperpartisan : supervision et limites

Les travaux récents sur la détection de biais/idéologie s'appuient majoritairement sur des corpus journalistiques écrits, avec des annotations plus fines (p. ex. *bias spans*) et des baselines Transformers. Des jeux de données comme BASIL ou Hyperpartisan ont contribué à opérationnaliser ces notions et à standardiser l'évaluation.

2.4.3 Robustesse, domain shift et facteurs confondants

Un risque méthodologique récurrent est le **domain shift** : un modèle peut "réussir" en exploitant des indices non désirés (source, style, longueur, thématiques) plutôt que l'idéologie au sens strict. La littérature OOD et l'adaptation de domaine (DAPT/TAPT) propose des protocoles et des stratégies pour réduire cet écart.

2.4.4 ASR -> NLP : faisabilité, bruit, propagation d'erreurs

L'ouverture à l'oral est désormais réaliste grâce à des ASR robustes (Whisper), mais la transcription introduit un bruit spécifique (ponctuation, entités, homophones). Cela justifie des analyses séparées "web vs ASR", et une instrumentation explicite (métriques par sous-corpus, matrices de confusion, contrôles anti-fuite).

2.4.5 Justifier le choix "corpus original + pipeline outillé"

Enfin, les travaux en *web corpus building* et extraction/normalisation soulignent que la validité dépend fortement de la chaîne amont (collecte, nettoyage, déduplication, métadonnées, reproductibilité). Dans notre cas, la contribution est moins un nouveau modèle qu'un **cadre expérimental** reproductible et extensible, conçu pour observer *où* se situe le signal (sémantique, style, cadrage) et comment il varie selon les domaines.

> La bibliographie est regroupée en fin de document.

3. Corpus

3.1 Vue d'ensemble

Deux corpus TEI :

- **WEB1** : corpus écrit issu d'un scraping massif (septembre 2025), consolidé en TEI (**514 340** documents ; **3.09 GiB** XML ; ≈ **352.36 M** tokens whitespace).
- **ASR1** : sous-corpus oral (YouTube), transcrit via Whisper et structuré en TEI (**1 000** documents ; **24.18 MiB** XML ; ≈ **4.07 M** tokens whitespace). Le corpus correspond à **821 médias** pour ≈ **352 h** d'audio (somme des durées).

Contraintes matérielles : exécutions sur CPU (Ryzen 7840U), avec forte sensibilité au temps de calcul. Cette contrainte motive l'usage de profils « quick » et de cap de taille pour certains entraînements.

3.1.1 Statistiques globales (audit TEI sur les corpus bruts)

Les chiffres ci-dessous proviennent d'un **audit TEI** réalisé sur les deux corpus *bruts* (avant filtrage `min_chars`, troncatures, caps, et sous-échantillonnage "quick"). Ils donnent un **ordre de grandeur** utile pour interpréter les contraintes CPU et le *domain shift*.

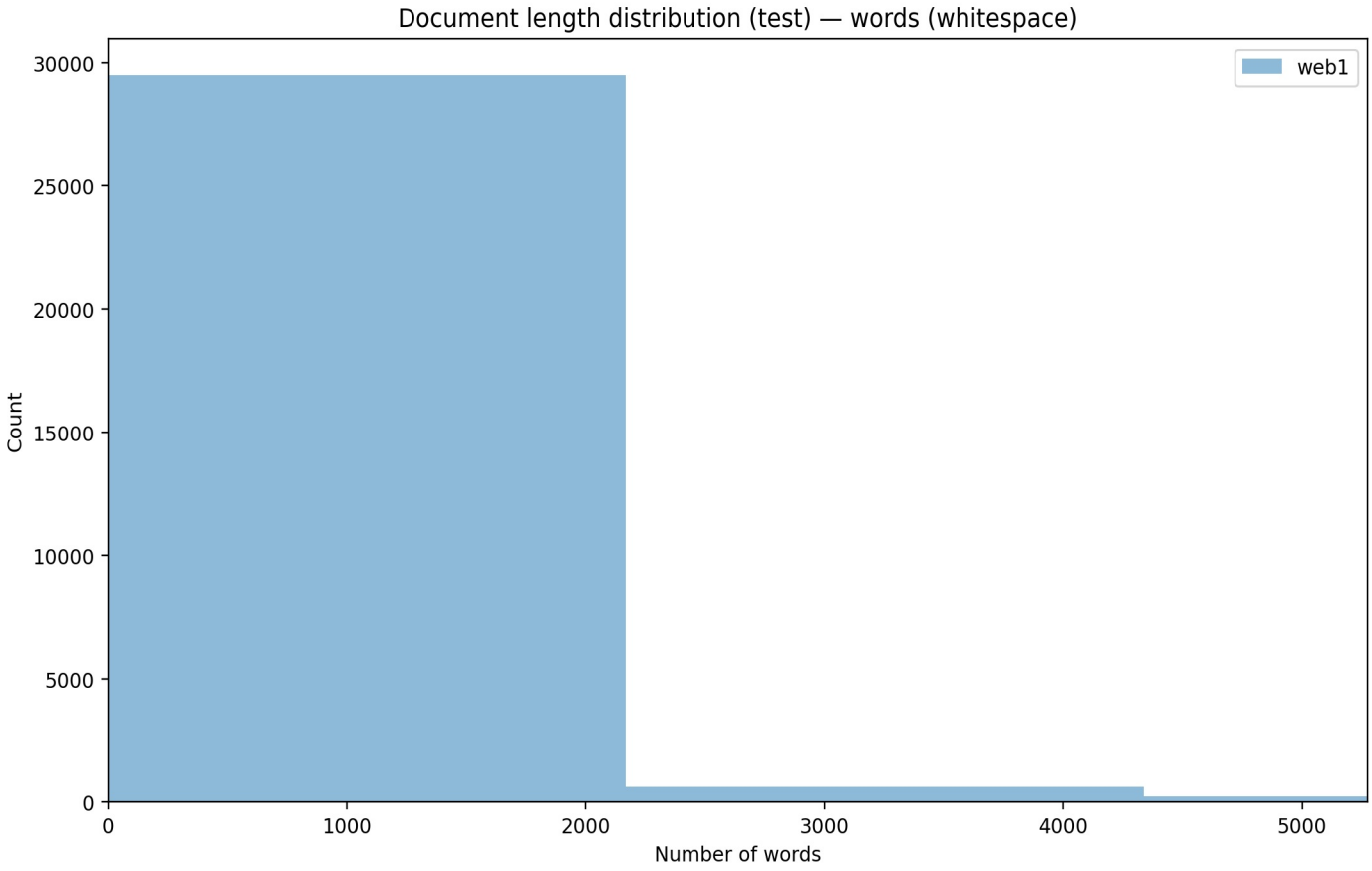
corpus	xml_size	gzip_size	docs	tokens	types_est	tok/doc_mean	tok/doc_p50	tok/doc_p95
web1 (web)	3.09 GiB	899.31 MiB	514 340	352 359 819	5 447 061	685,1	295	2 348
asr1 (audio→ASR)	24.18 MiB	7.67 MiB	1 000	4 067 385	179 114	4 067,4	1 158	17 335

Notes méthodologiques (lecture correcte des chiffres)

- Les **tokens** sont estimés par **tokenisation whitespace** (proxy de volumétrie, comparable entre corpus, mais différent d'une tokenisation linguistique).
- `types_est` est une estimation de vocabulaire (ordre de grandeur), utile pour discuter la diversité lexicale.
- Les expériences rapportées ensuite portent sur des **datasets dérivés** (sous-ensembles filtrés et parfois capés), donc la taille effective vue par les modèles peut être très inférieure au corpus brut.
- Pour **recalculer** ces statistiques sur un autre instantané de corpus : `make audit_tei CORPUS_ID=web1` / `make audit_tei CORPUS_ID=asr1` (l'audit utilise une tokenisation whitespace pour fournir un proxy de volumétrie).

3.1.2 Longueur des documents (effet pratique sur la modélisation)

La longueur des documents influence directement (i) le **filtrage** (`min_chars`, `max_tokens`), (ii) la **troncature** (notamment pour les modèles modernes), et (iii) la stabilité des représentations (TF-IDF). La figure suivante illustre la distribution des longueurs observées dans un dataset de préparation “idéologie” (dans l’archive de résultats, ce dataset ne contient que des documents web).



3.2 WEB1 - collecte, extraction, structuration

3.2.1 Sources (20 sites)

Le noyau initial est constitué de 20 sites (organisations politiques et médias) afin d’obtenir un spectre varié.

Site	label_raw	Orientation
actionfrancaise	crawl_actionfrancaise_20251003_000000	droite
cce_full	crawl_cce_full_20250927_234505	droite
contretemps	crawl_contretemps_20250928_003525	gauche
eco_full	crawl_eco_full_20250928_000354	gauche
er	crawl_er_20251003_231234	droite
fdesouche	crawl_fdesouche_20251003_231247	droite
hut_fr	crawl_hut_fr_20251003_231255	droite
initiative_communiste	crawl_initiative_communiste_20250928_003234	gauche
jean_jaures	crawl_jean_jaures_20250928_003513	gauche
lfi_full	crawl_lfi_full_20250928_000133	gauche
lo_full	crawl_lo_full_20250928_001323	gauche
lr_full	crawl_lr_full_20250928_001404	droite
lundi_am	crawl_lundi_am_20250928_003202	gauche
npa_full	crawl_npa_full_20250928_001251	gauche
polemia	crawl_polemia_20250928_003004	droite
ps_full	crawl_ps_full_20250928_000302	gauche
revperm	crawl_revperm_20250928_003219	gauche
rn_full	crawl_rn_full_20250927_234616	droite

Site	label_raw	Orientation
terranova	crawl_terranova_20250928_003520	gauche
ucl	crawl_ucl_20250928_003510	gauche

3.2.2 Chaîne de collecte et extraction

1) Collecte d'URLs et crawl (outil Minet / médialab), filtrage automatique d'URLs cassées. 2) Fetch massif : stockage en arborescence de HTML compressés + CSV de crawl. 3) Extraction texte : Trafilatura (choix volontaire d'un outil unique pour limiter la complexité sous contrainte de temps). 4) Consolidation : regroupement par classes/sources puis export TEI.

3.2.3 TEI minimaliste et fiable (compromis)

La conversion brut -> TEI a été effectuée par un script pragmatique fondé sur des heuristiques (regex sur chemins/dossiers). Cela n'est pas « élégant », mais ce compromis a permis de produire rapidement un TEI stable. Une partie des métadonnées riches est restée dans les archives brutes (CSV/HTML) plutôt que d'être injectée dans le TEI.

3.2.4 Volumétrie et distributions (WEB1, audit)

Sur la base de l'audit TEI (corpus brut), WEB1 contient **514 340** documents pour ≈ **352.36 M** tokens whitespace (médiane **295** tokens/doc ; p95 **2 348**). Cette volumétrie impose (i) des profils "quick" et (ii) des stratégies de filtrage et de cap.

Langues (top) : fr=485 746, en=18 359, es=3 574, de=1 858, it=1 109

Concentration par domaines (top 10, nombre de documents)

Domaine (top 10)	Docs
fdesouche.com	178 130
egaliteetreconciliation.fr	46 518
revolutionpermanente.fr	34 571
initiative-communiste.fr	20 778
polemia.com	8 560
contretemps.eu	8 219
actionfrancaise.net	7 361
lafranceinsoumise.fr	7 320
lundi.am	5 839
jean-jaures.org	5 826

Concentration par crawls (top 10, nombre de documents)

Crawl (top 10)	Docs
crawl-fdesouche-20251003_231247	184 500
crawl-er-20251003_231234	77 155
crawl-lo-full-20250928_001323	62 716
crawl-revperm-20250928_003219	58 081
crawl-initiative-communiste-20250928_003234	33 823
crawl-contretemps-20250928_003525	18 742
crawl-polemia-20250928_003004	13 207
crawl-lundi-am-20250928_003202	12 325
crawl-jean-jaures-20250928_003513	11 902
crawl-ucl-20250928_003510	11 501

Indicateurs opérationnels de collecte (archives brutes, ordre de grandeur)

- jobs total : **1 067 977** ; 2xx=90.1%, 4xx=9.2%, 5xx=0.1%, erreurs=0.2%
- fichiers HTML collectés : **1 682 743** (volume ≈ **33.8 GiB**)

Ces distributions sont méthodologiquement importantes : elles matérialisent un risque de **signal "source"** et de **corrélations label média**, ce qui motive (i) la tâche diagnostique `crawl`, et (ii) la prudence interprétative sur la tâche idéologie.

3.3 ASR1 - YouTube -> Whisper -> TEI

3.3.1 Choix du seuil (1000 audios)

La collecte YouTube est coûteuse techniquement sans API. Le projet s'est arrêté à **1000 audios** après mesure : ≈ **352 h** d'audio, jugées suffisantes pour un sous-corpus oral, tout en restant compatible avec les délais.

3.3.2 Transcription

- Audio : yt-dlp
- ASR : Whisper (base), exécution CPU (≈ 10 jours cumulés)
- Structuration : JSON -> TEI

3.3.3 Volumétrie et répartition (ASR1, audit)

Au niveau TEI (corpus brut), ASR1 contient **1 000** documents pour **≈ 4.07 M** tokens whitespace (médiane **1 158** tokens/doc ; p95 **17 335**), ce qui reflète le passage à un registre oral transcrit (séquences longues, ponctuation et segmentation moins stables).

Au niveau *médias*, le corpus a été construit à partir de **821 fichiers audio/vidéo**, totalisant **≈ 352 h** (somme des durées). Une estimation initiale plus haute (≈ 500 h) a circulé au cours du projet ; le chiffre retenu ici correspond au décompte des médias effectivement conservés dans la version consolidée.

Répartition indicative par `asr_party` (top 10, nombre de documents)

asr_party (top 10)	docs
asr_party:er	369
asr_party:fdesouche	169
asr_party:revperm	169
asr_party:lundi-am	118
asr_party:ucl	44
asr_party:action_francaise	33
asr_party:npa	30
asr_party:jean-jaures	22
asr_party:eco	13
asr_party:terra_nova	10

3.4 Labels et tâches

3.4.1 Ideology_global (binaire)

Définition opérationnelle : **gauche/droite** basée sur l'idéologie globale de l'acteur/source. Tâche conçue comme baseline simple et stable.

3.4.2 Crawl (multi-classes ~20)

Tâche multi-classes au niveau du crawl/source. Elle sert de diagnostic : mesurer la prédictibilité de la provenance et la « signature » éditoriale.

4. Méthode

Le sujet exige une chaîne complète : préparation -> représentation -> apprentissage -> évaluation, avec comparaison de méthodes et analyse critique des résultats.

4.1 Architecture et commande universelle

Le pipeline est piloté par profils YAML + une commande universelle (via `make run` et des overrides), afin de :

- reproduire une expérience à l'identique,

- changer un paramètre sans dupliquer des scripts,
- lancer des analyses en parallèle (familles de modèles) tout en gardant les artefacts séparés.

4.2 Séparation des stages (prepare/train/evaluate)

La séparation des phases a été choisie car ce sont des blocs historiquement instables et indépendants :

- `prepare` (TEI -> TSV, filtrage, split, dédup),
- `train` (apprentissage),
- `evaluate` (métriques, rapports).

Limite assumée : l'évolution des exigences a entraîné une centralisation sur quelques scripts longs ; une factorisation plus fine est laissée en perspective.

4.3 Configurations effectives des runs `FINAL_*` (stabilisées)

> Les paramètres ci-dessous sont extraits des logs d'exécution et définissent les **conditions expérimentales** des runs `FINAL_*`.

4.3.1 Ideology_global - WEB1 (dataset_id = `FINAL_web1_ideo`)

- **Profil** : `ideo_quick` (famille `sklearn`, modèle `tfidf_svm_quick`)
- **Préparation** : `tokenizer=split`, `dedup_on=text`, `min_chars=200`, `max_tokens=512`, `seed=52`
- **Balance** : `class_weights` (train ré-pondéré, dev/test naturels)
- **Splits** : train=198242, dev=24780, test=24781
- **Déduplication** : 247947 -> 247803 (sur le texte)
- **Troncature train sklearn** : `max_train_docs_sklearn=100000` (198242 -> 100000)
- **Artefacts attendus** :
  - `reports/FINAL_web1_ideo/ideology_global/sklearn/tfidf_svm_quick/metrics.json`
  - `reports/FINAL_web1_ideo/ideology_global/sklearn/tfidf_svm_quick/classification_report.txt`

4.3.2 Ideology_global - ASR1 (ASR only)

Dans l'archive `reports/` jointe au rendu, **aucun run `FINAL_asr1_ideo` n'est présent**. Le run **ASR-only disponible** est un run de diagnostic :

- **Run** : `diag_asr1_ideo_base/ideology_global/sklearn/tfidf_svm_quick`
- **Évaluation** : n_test = 94, accuracy = 0.755, macro-F1 = 0.745

Ce résultat est utilisé dans le rapport comme **diagnostic** (support faible), pour objectiver la baisse de performance sur l'oral transcrit et motiver la présentation systématique des métriques **par sous-corpus** et **balanced**.

4.3.3 Ideology_global - Mix WEB1+ASR1 (comparaison multimodale)

Deux dossiers de résultats existent dans `reports/` :

1) `FINAL_mix_ideo/ideology_global/sklearn/tfidf_svm_quick` : dans cette archive, `metrics_by_corpus_id.json` **ne contient que `web1`** (l'ASR n'est pas inclus). Ce run est donc interprété comme un run "web" (malgré son nom) et n'est **pas** utilisé pour la comparaison web vs ASR.

2) `web1_asr1/ideology_global/sklearn/tfidf_svm_quick` : run "mix" **effectif**, utilisé pour la comparaison multimodale.

- métriques globales : `reports/web1_asr1/ideology_global/sklearn/tfidf_svm_quick/metrics.json`
- métriques par corpus : `reports/web1_asr1/ideology_global/sklearn/tfidf_svm_quick/metrics_by_corpus_id.json`
- métriques balanced (poids 0.5/0.5) : `reports/web1_asr1/ideology_global/sklearn/tfidf_svm_quick/metrics_balanced_by_corpus_id.json`

C'est ce second run qui alimente la section Résultats (domain shift web -> ASR).

Paramètres effectifs :

- **Profil** : `ideo_quick_web1_asr1` (famille `sklearn`)
- **Préparation** : `tokenizer=split`, `dedup_on=text`, `min_chars=200`, `max_tokens=512`, `seed=time`
- **Balance** : `class_weights`
- **Splits** : train=173675, dev=37216, test=37216 (train_prop=0.7)
- **Déduplication** : 248254 -> 248107
- **Troncature train sklearn** : 173675 -> 100000
- **Évaluation** : production de métriques globales + **métriques par corpus_id** et **balanced_by_corpus_id**.
    - `reports/web1_asr1/ideology_global/sklearn/tfidf_svm_quick/metrics.json`
    - `reports/web1_asr1/ideology_global/sklearn/tfidf_svm_quick/metrics_by_corpus_id.json`
    - `reports/web1_asr1/ideology_global/sklearn/tfidf_svm_quick/metrics_balanced_by_corpus_id.json`

4.3.4 Crawl multi-classes - WEB1 (dataset_id = `FINAL_web1_crawl`)

- **Profil** : `custom` (vue `crawl`, `label_field=crawl`, famille `sklearn`)
- **Préparation** : `tokenizer=whitespace`, `dedup_on=none`, `min_chars=400`, `max_tokens=512`, `seed=42`
- **Balance** : `cap_docs` avec preset `capdocs_crawl_3k` (cap par classe)
- **Anti-fuite inter-splits** : suppression de doublons (text-hash) : train=140, dev=25, test=41
- **Splits naturels** (avant cap) : train=265711, dev=33207, test=33189
- **Train final après cap** : 40274 docs
- **Modèles entraînés** :
- `tfidf_svm_quick`, `tfidf_smo_linear`, `tfidf_smo_rbf`, `tfidf_perceptron`, `tfidf_randomtree`, `tfidf_randomforest`

4.4 Multimodalité (web vs ASR) et filtrage des documents

Le pipeline a été conçu pour comparer des sous-corpus **écrits (web)** et **oraux transcrits (ASR)**, avec une instrumentation explicite (métriques par sous-corpus, figures, anti-fuite). Dans notre corpus :

- **web1** : corpus web issu d'un scraping (articles/pages), extraction de texte via *trafilatura*.
- **asr1** : corpus de **1000 audios** transcrits via Whisper (≈ 352h), donc plus hétérogène au niveau syntaxique et discursif.

4.4.1 Filtrage : paramètres effectifs (profil `ideo_quick_asr1`)

Le profil ASR "quick" applique un filtrage minimaliste mais strict, afin de garantir (i) la **qualité** des instances, (ii) des temps de calcul raisonnables, et (iii) une stabilité des splits :

- longueur minimale : `min_chars = 200`
- longueur maximale : `max_tokens = 512`
- tokenisation : `tokenizer = split`
- déduplication : `dedup_on = text`
- plafond global "quick" : `max_docs_global = 10000`
- politique "unknown actors" : `drop` (via `actors_yaml`)

Zone d'ombre à objectiver : lors d'un run "ASR only" après nettoyage complet (`make clean`), la phase `prepare` peut aboutir à **0 documents valides** ("Aucun document valide après filtrage"). L'hypothèse la plus probable est un effet combiné du seuil `min_chars` et de la segmentation/qualité des transcriptions, mais ce point doit être quantifié (voir § 6, "analyses minimales à finaliser").

4.4.2 Stratégie de présentation : ASR instrumenté via le run "mix"

Compte tenu des contraintes matérielles/temps, nous privilégions une présentation robuste : (i) **web1 only** comme baseline "stable", puis (ii) un run **mix web1+asr1** avec métriques par sous-corpus (web vs ASR). Cette stratégie permet de quantifier immédiatement le **domain shift** : même à labels identiques (idéologie binaire), les performances diffèrent selon la modalité (style, bruit ASR, structure discursive).

4.5 Évaluation et métriques

Conformément au sujet : exactitude, précision, rappel, F1, matrices de confusion, et analyse d'erreurs. Le mix web+ASR exige en plus des métriques **par sous-corpus** et **balanced-by-corpus**.

5. Résultats

Cette section présente (i) les **runs finals** produits par le pipeline (dossiers `reports/FINAL_*`) et (ii) quelques **runs de validation** issus d'analyses post-hoc (artefacts `cm_*`, `summary_*`) permettant d'ajouter des matrices de confusion et de faciliter l'analyse d'erreurs. Les métriques reportées sont celles du **jeu de test**. Les figures incluses ci-dessous sont limitées à celles qui restent lisibles (binaire, comparatifs macro).

5.1 Résultats principaux

Cette section regroupe les résultats les plus exploitables et lisibles issus des runs "finalisés". Les métriques détaillées (rapports de classification, JSON de métriques, matrices de confusion) sont archivées dans `reports/` et référencées dans le dépôt.

5.1.1 Idéologie binaire (baseline) - TF-IDF + SVM (sklearn)

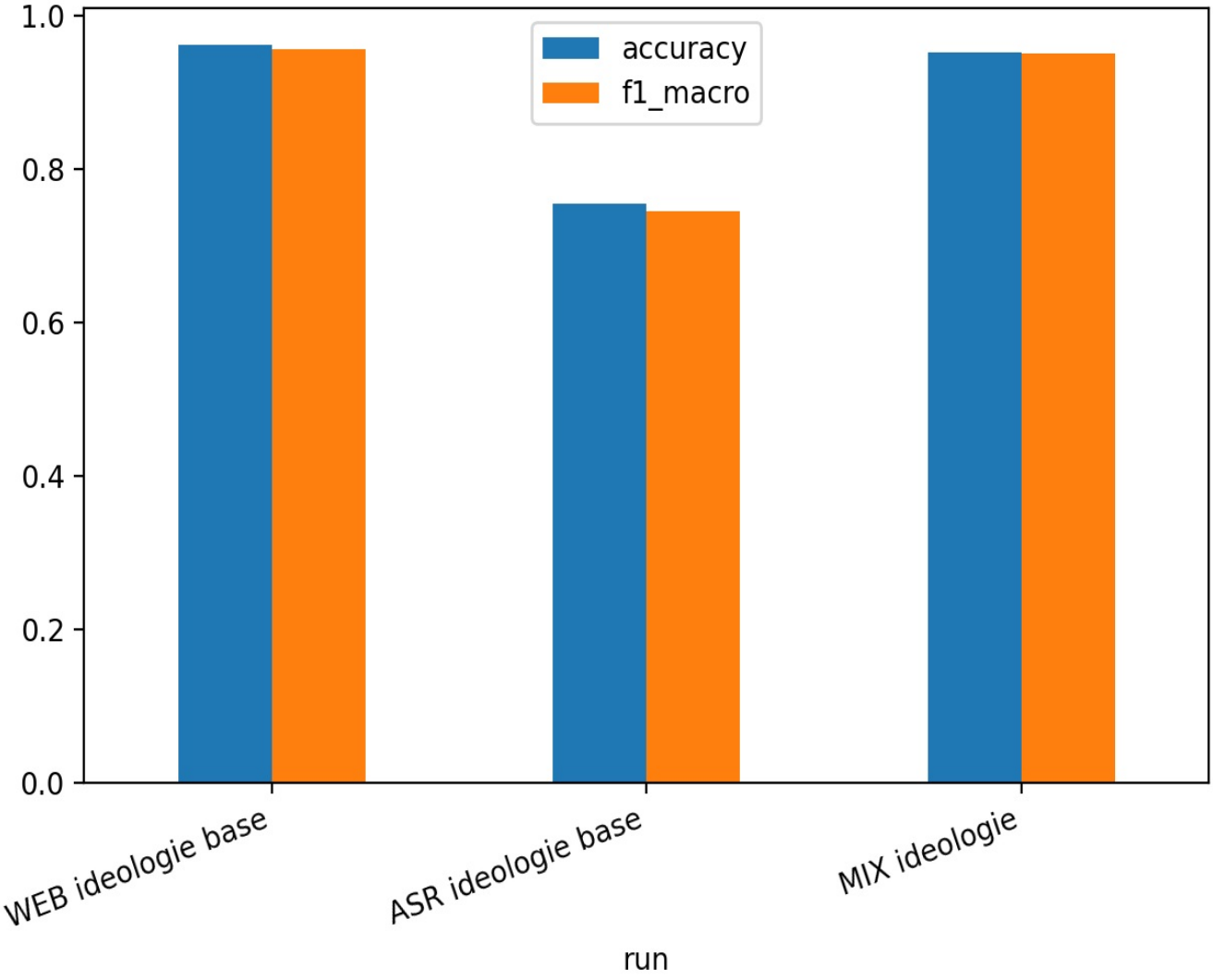
Tâche : prédire `left` / `right` (idéologie globale), avec un protocole train/dev/test stratifié. **Modèle :** `tfidf_svm_quick` (SVM linéaire sur TF-IDF), choisi comme baseline robuste, rapide et interprétable.

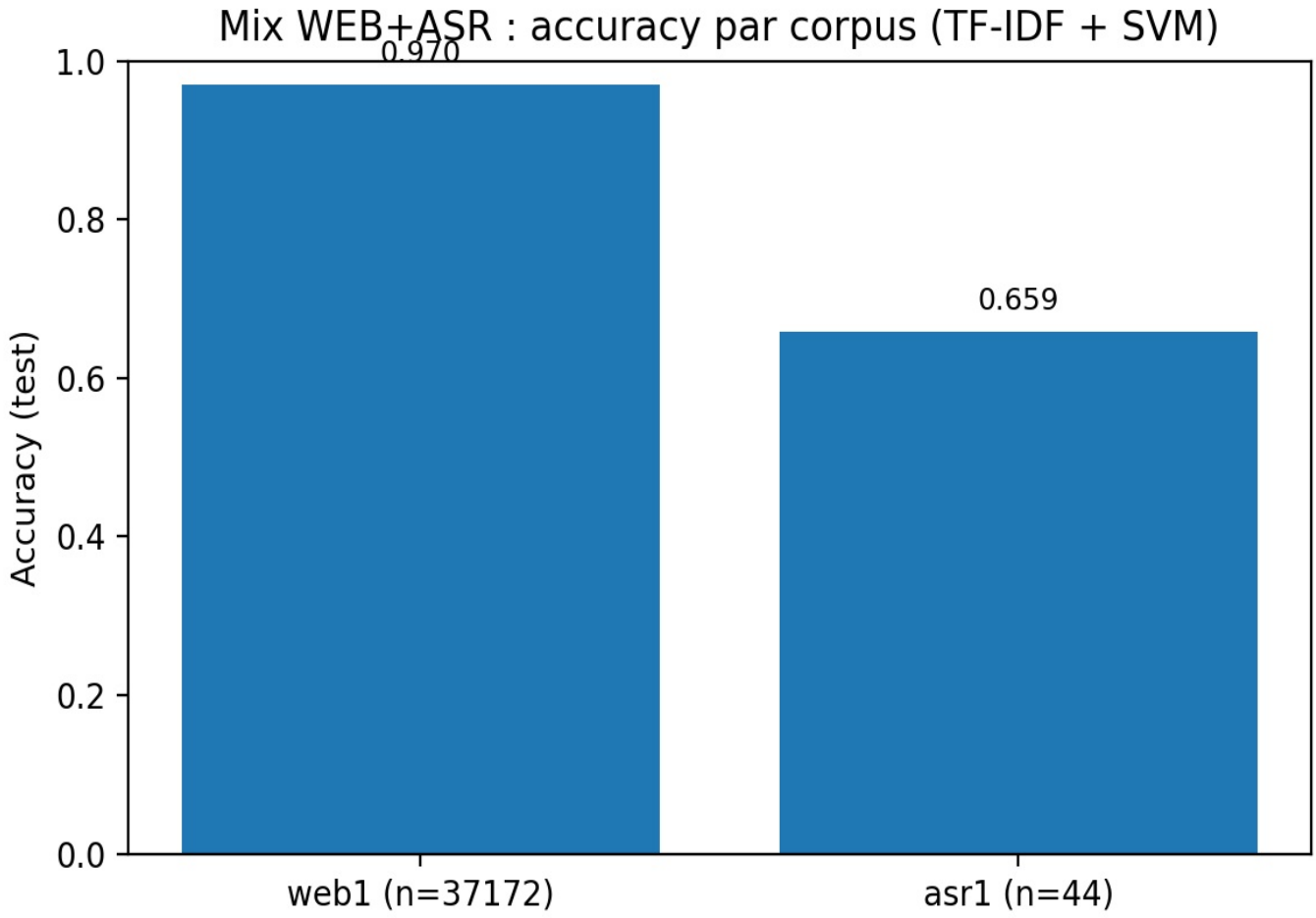
Run (référence reports/)	n_test	Accuracy	Macro-F1	Notes
WEB1 only (`FINAL_web1_ideo`, `tfidf_svm_quick`)	24781	0.972	0.967	Baseline écrite, gros volume, stable
ASR1 only (`diag_asr1_ideo_base`, `tfidf_svm_quick`)	94	0.755	0.745	Diagnostic (support faible, n=94)
MIX web1+asr1 (global) (`web1_asr1`, `tfidf_svm_quick`)	37216	0.972	0.967	Dominé par web1 (déséquilibre de volume)
└── sous-corpus **web1**	37173	0.972	0.967	Proche de WEB-only
└── sous-corpus **asr1**	43	0.721	0.609	Domain shift + bruit ASR
MIX (balanced par corpus) (poids 0.5/0.5)	-	0.846	0.788	Lecture robuste du domain shift

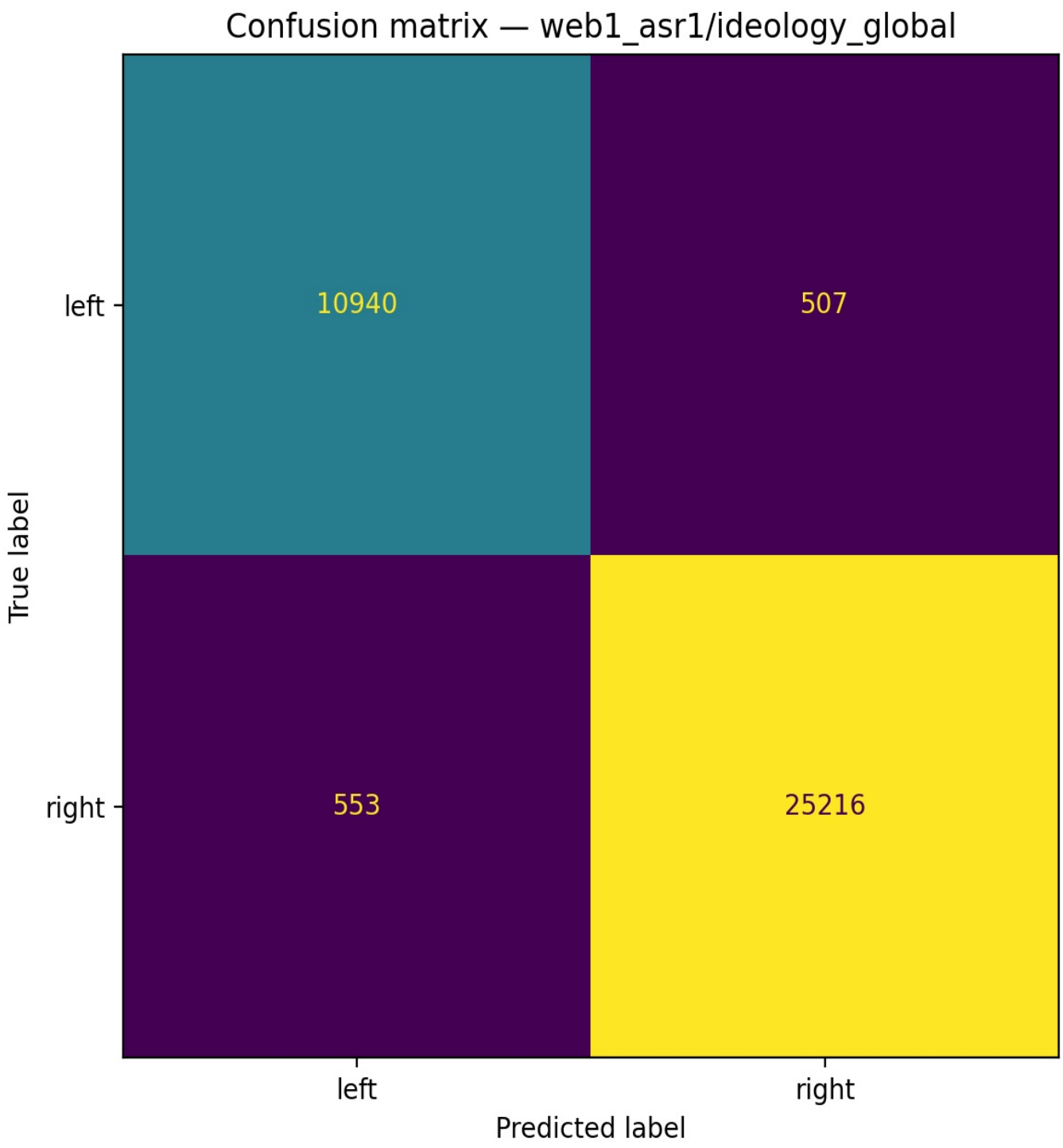
Lecture :

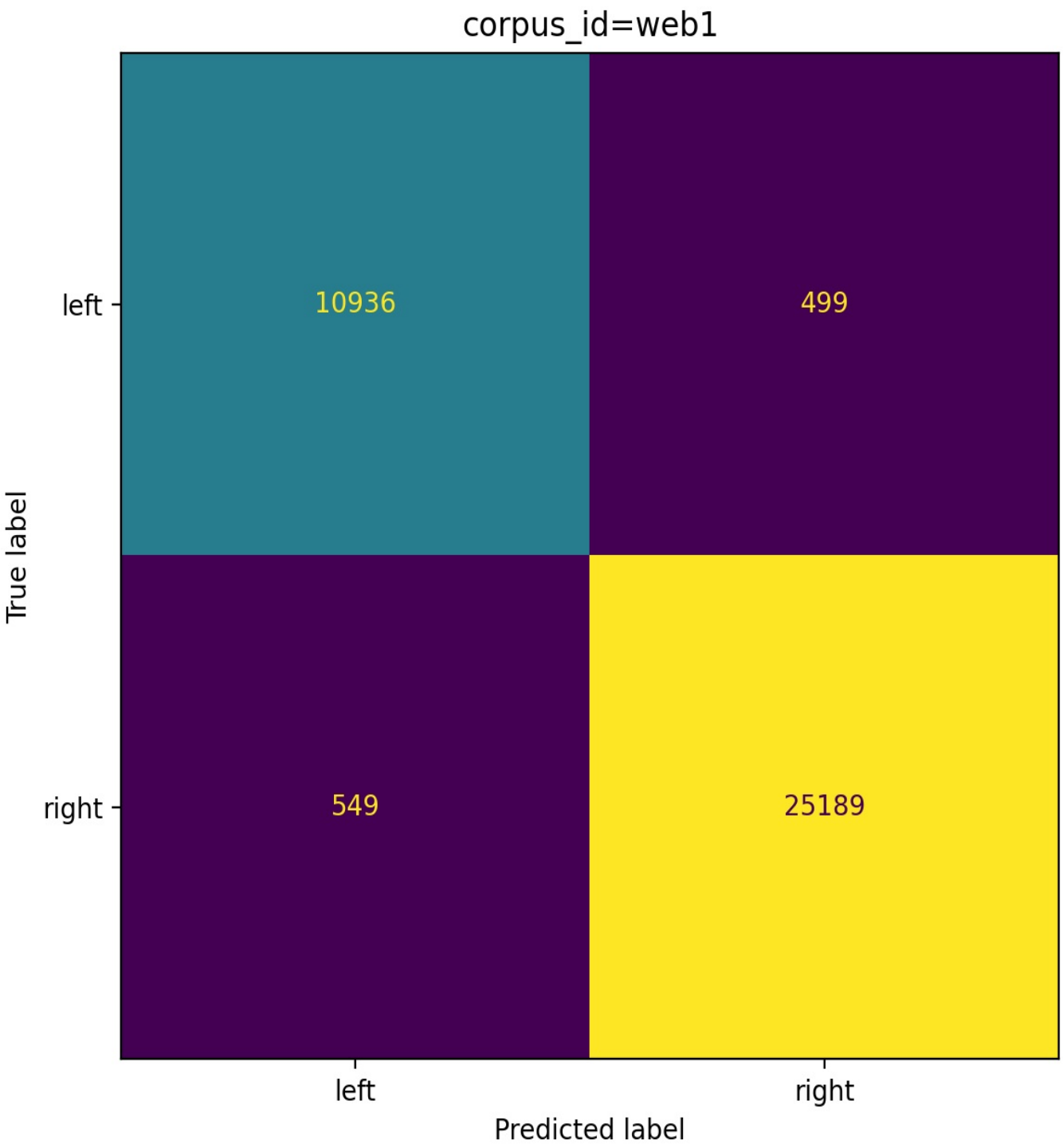
- `web1 only` fournit une baseline "facile" (structure et signatures éditoriales plus homogènes) et sert de référence.
- le run `mix` permet une mesure directe du **domain shift** via les métriques par sous-corpus.
- la baisse observée sur **asr1** est cohérente avec un cumul de facteurs : bruit ASR, oralité, segmentation, et changement de structure discursive.

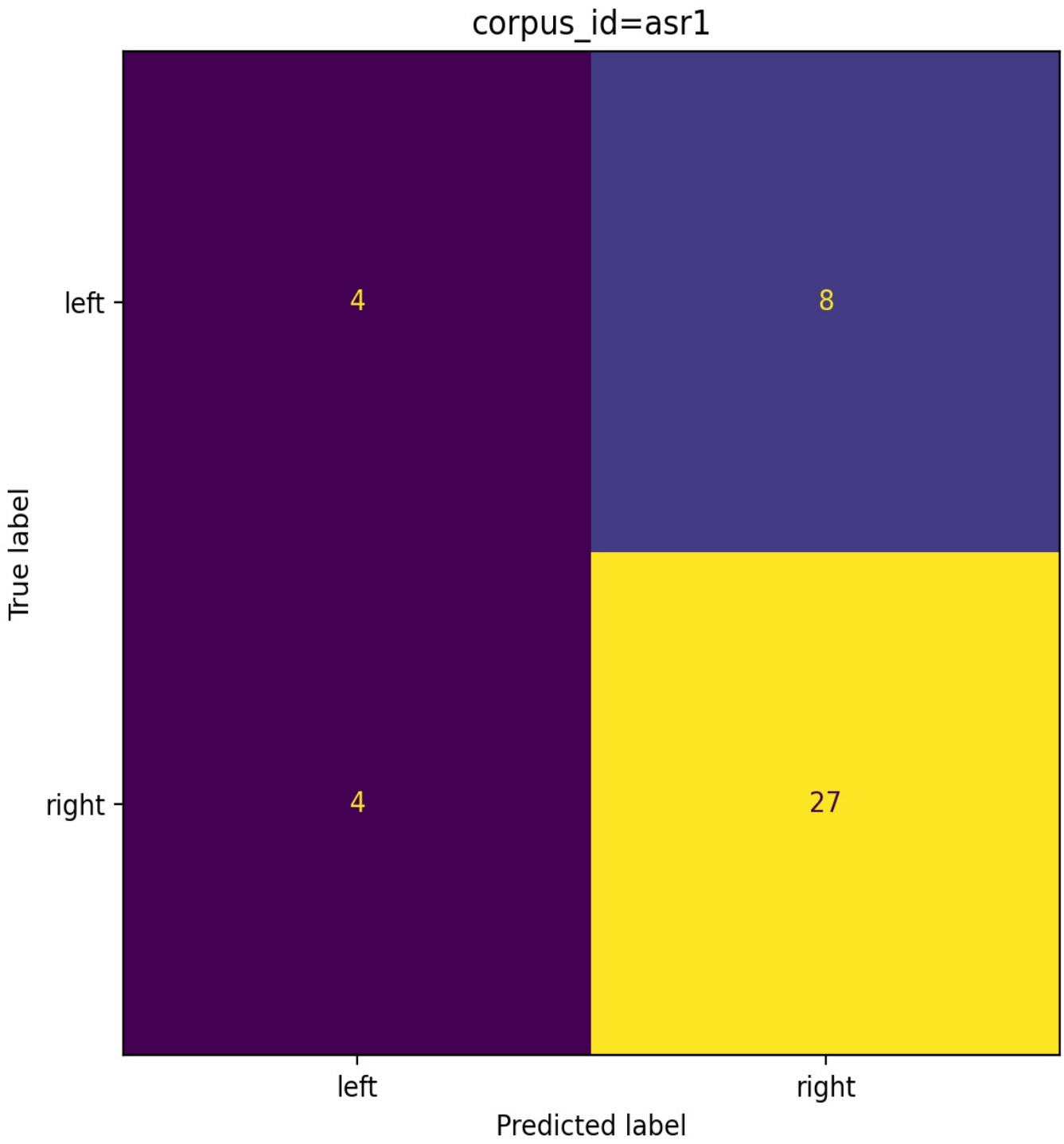
Figures associées (exploitables) :


1.0
0.8
0.6
0.4
0.2
0.0
accuracy
f1_macro
WEB ideologie base
ASR ideologie base
MIX ideologie
run


Mix WEB+ASR : accuracy par corpus (TF-IDF + SVM)
Accuracy (test)
1.0
0.8
0.6
0.4
0.2
0.0
0.970
0.659
web1 (n=37172)
asr1 (n=44)


Confusion matrix — web1_asr1/ideology_global
left
right
True label
10940
507
553
25216
left
right
Predicted label


corpus_id=web1
left
right
10936
499
549
25189
True label
left
right
Predicted label



5.1.2 Classification multi-classes (vue `crawl`) - signatures de source

Tâche : prédire la classe `crawl_*` (≈ 20 sources). **Intérêt scientifique dans PEPM :** cette tâche sert d'outil de diagnostic pour quantifier l'empreinte **source/ligne éditoriale** (et donc la part "style/provenance") dans le signal appris.

5.1.2.1 WEB1 (run final) - comparaison de plusieurs modèles sklearn

Pour rendre l'expérience faisable et comparable sur CPU, le run final applique un plafonnement d'entraînement (`cap_docs` par classe, preset de type "3k par crawl") et entraîne plusieurs modèles TF-IDF.

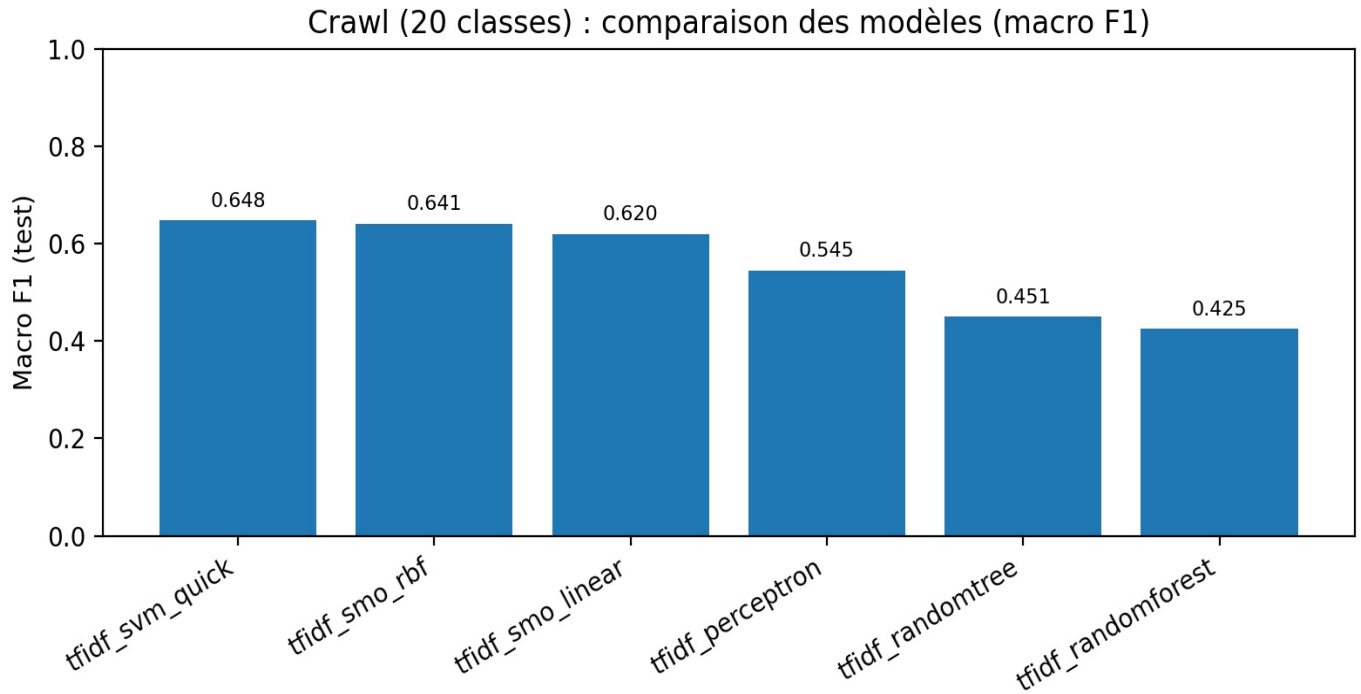
Modèle (FINAL_web1_crawl)	n_test	Accuracy	Macro-F1
tfidf_smo_rbf	51364	0.755	0.644
tfidf_svm_quick	51364	0.763	0.637
tfidf_smo_linear	51364	0.756	0.609
tfidf_perceptron	51364	0.681	0.571

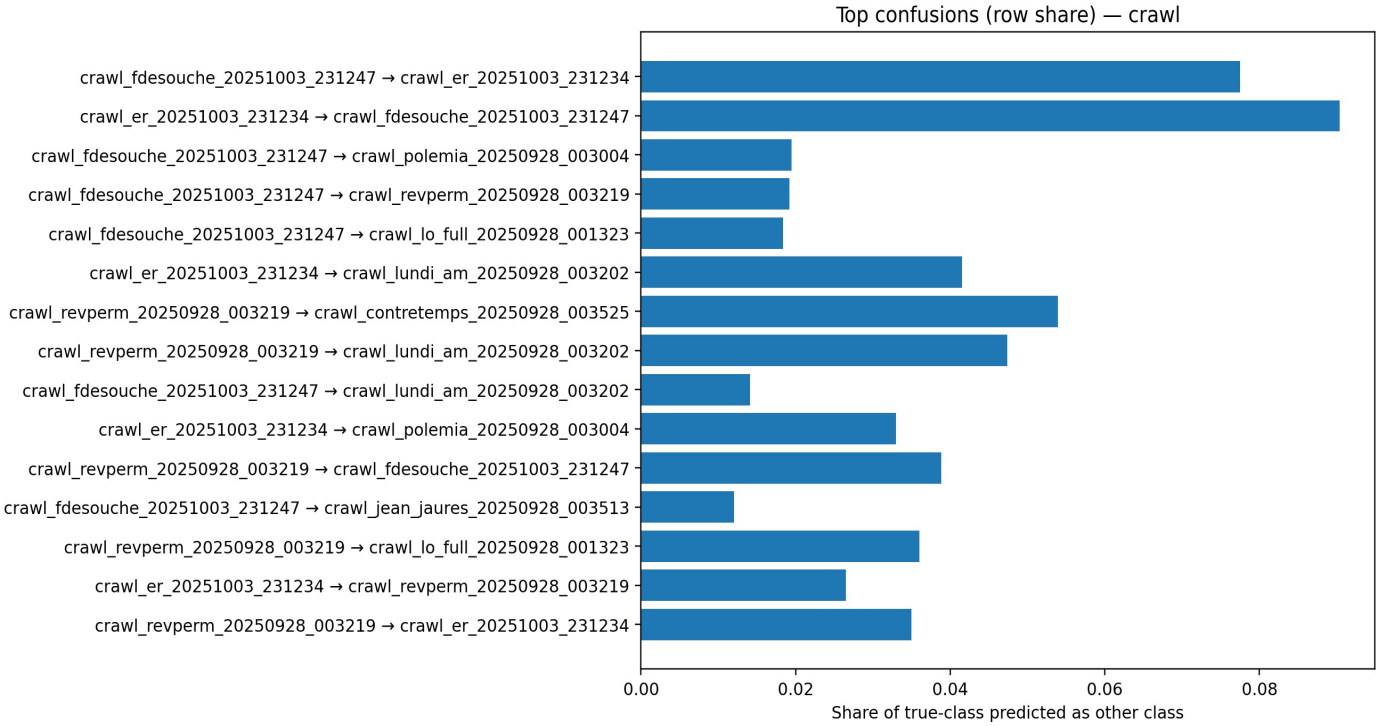
Modèle (FINAL_web1_crawl)	n_test	Accuracy	Macro-F1
tfidf_randomforest	51364	0.623	0.481
tfidf_randomtree	51364	0.442	0.455

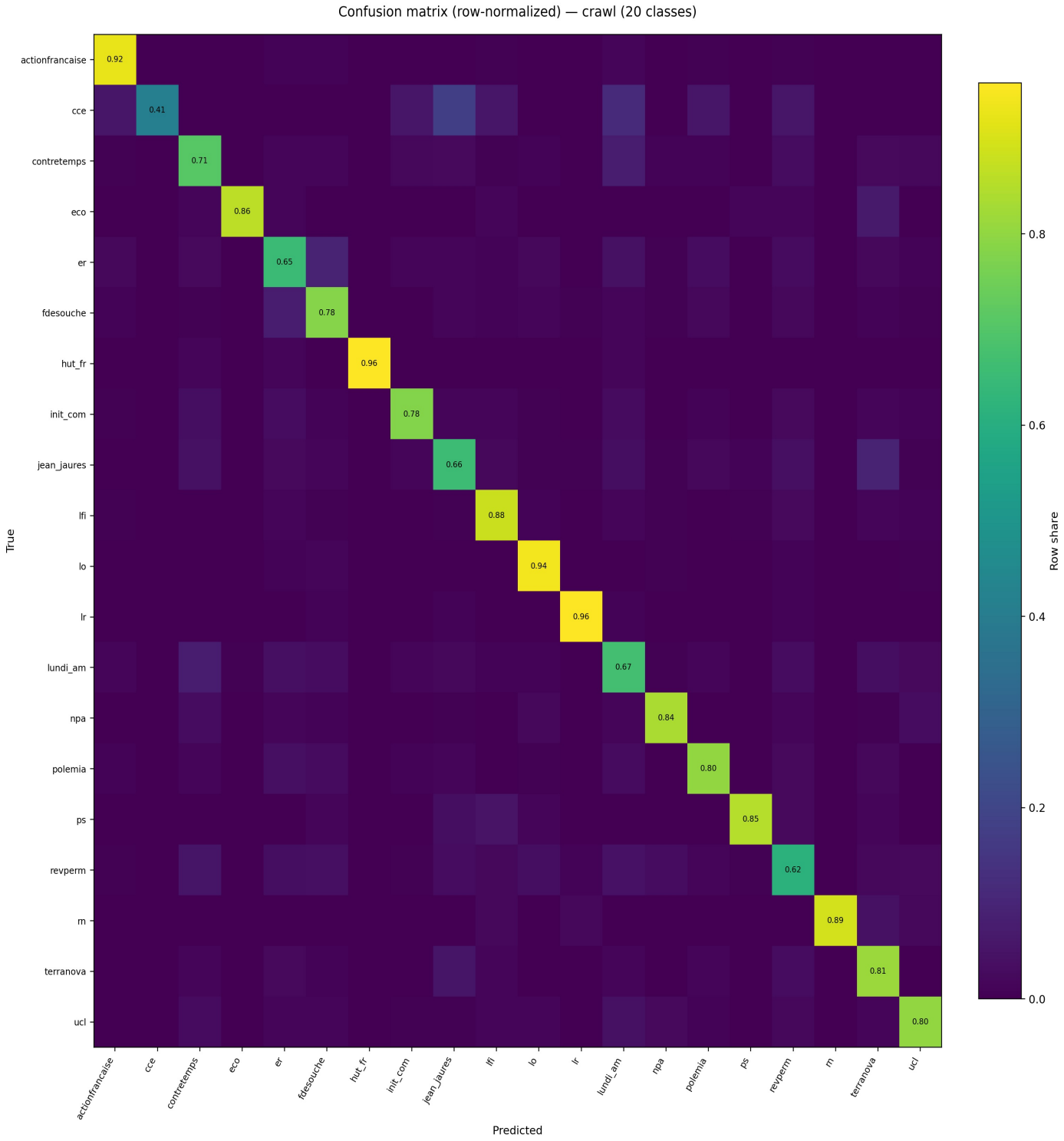
Lecture :

- La **macro-F1** est le bon indicateur ici (multi-classes déséquilibré).
- Le meilleur score macro-F1 est obtenu par **tfidf_smo_rbf** (SMO RBF) : cela suggère qu'une frontière non-linéaire capte mieux certaines confusions inter-sources, mais au prix d'un modèle moins interprétable.

Figures associées :







5.1.2.2 Mix web1+ASR1 (run diagnostic) - visualisation des confusions

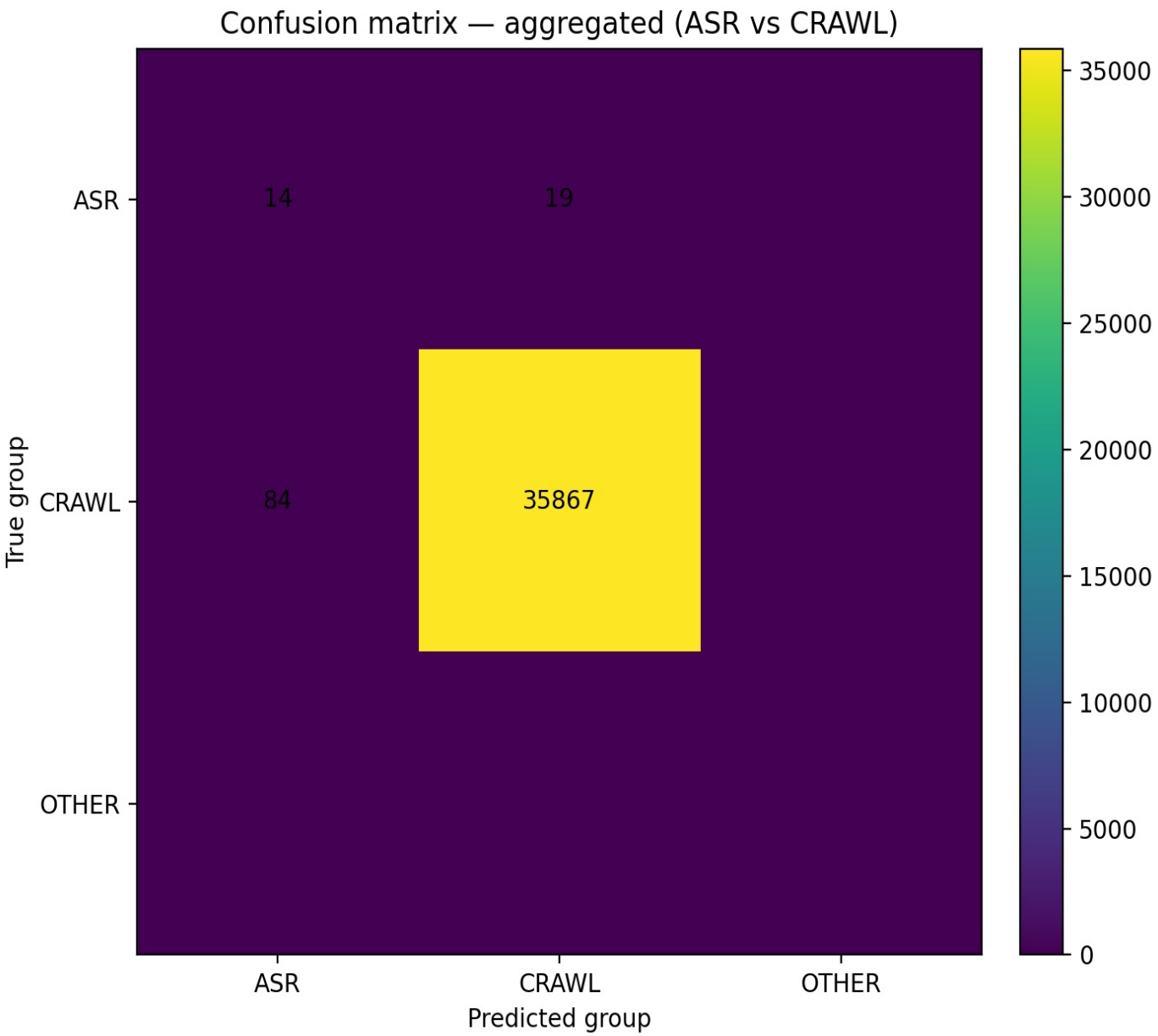
En complément, un run “mix” rapide (`web1_asr1/crawl/tfidf_svm_quick`) a été utilisé pour produire des **figures lisibles** de confusions (agrégations). Ce run n’est pas l’axe principal du rendu, mais sert à illustrer qualitativement la difficulté du multi-classes.

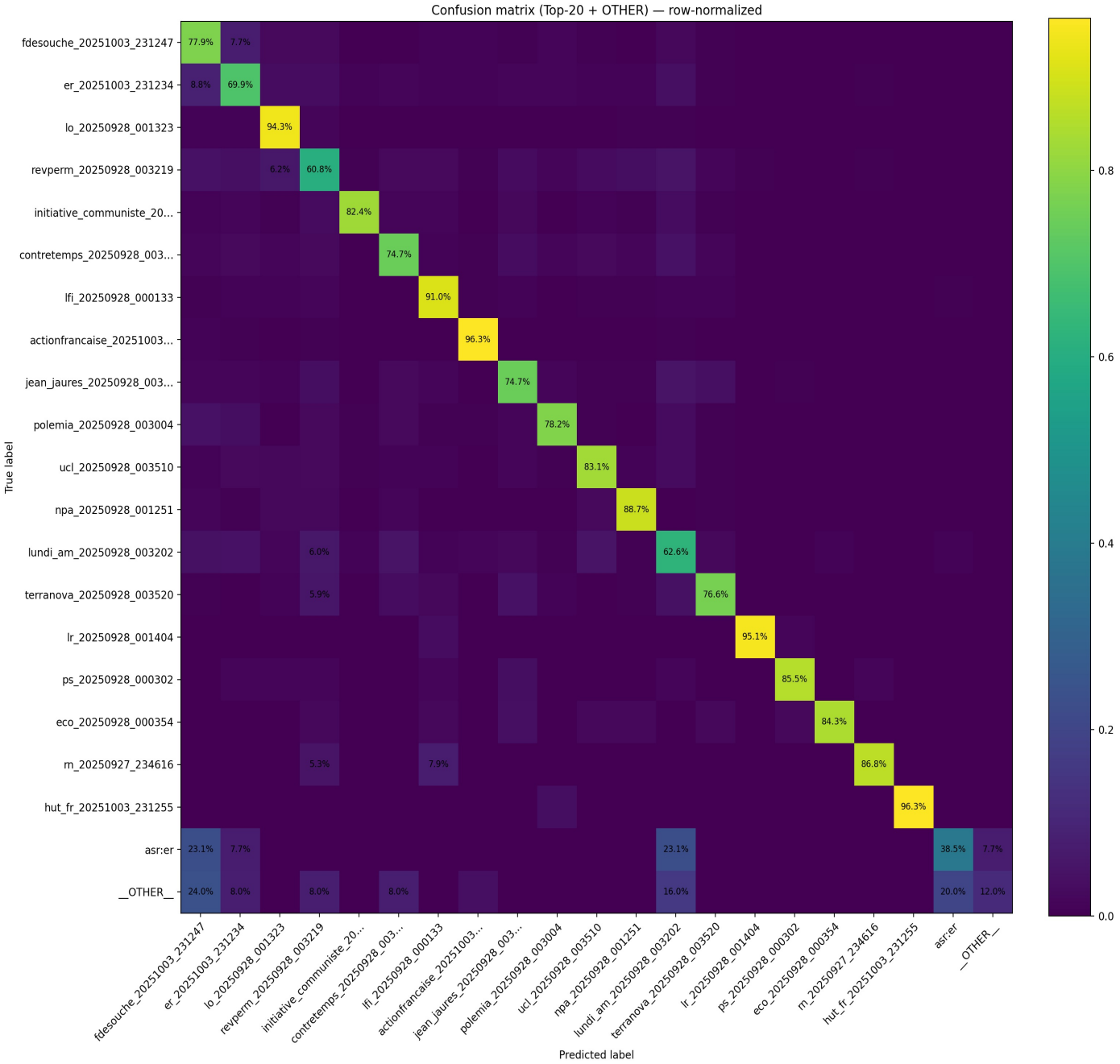
Run (diagnostic)	n_test	Accuracy	Macro-F1	Remarque
web1_asr1 / crawl / tfidf_svm_quick	35984	0.786	0.491	Accuracy élevée mais macro-F1 faible (multi-classes déséquilibré)

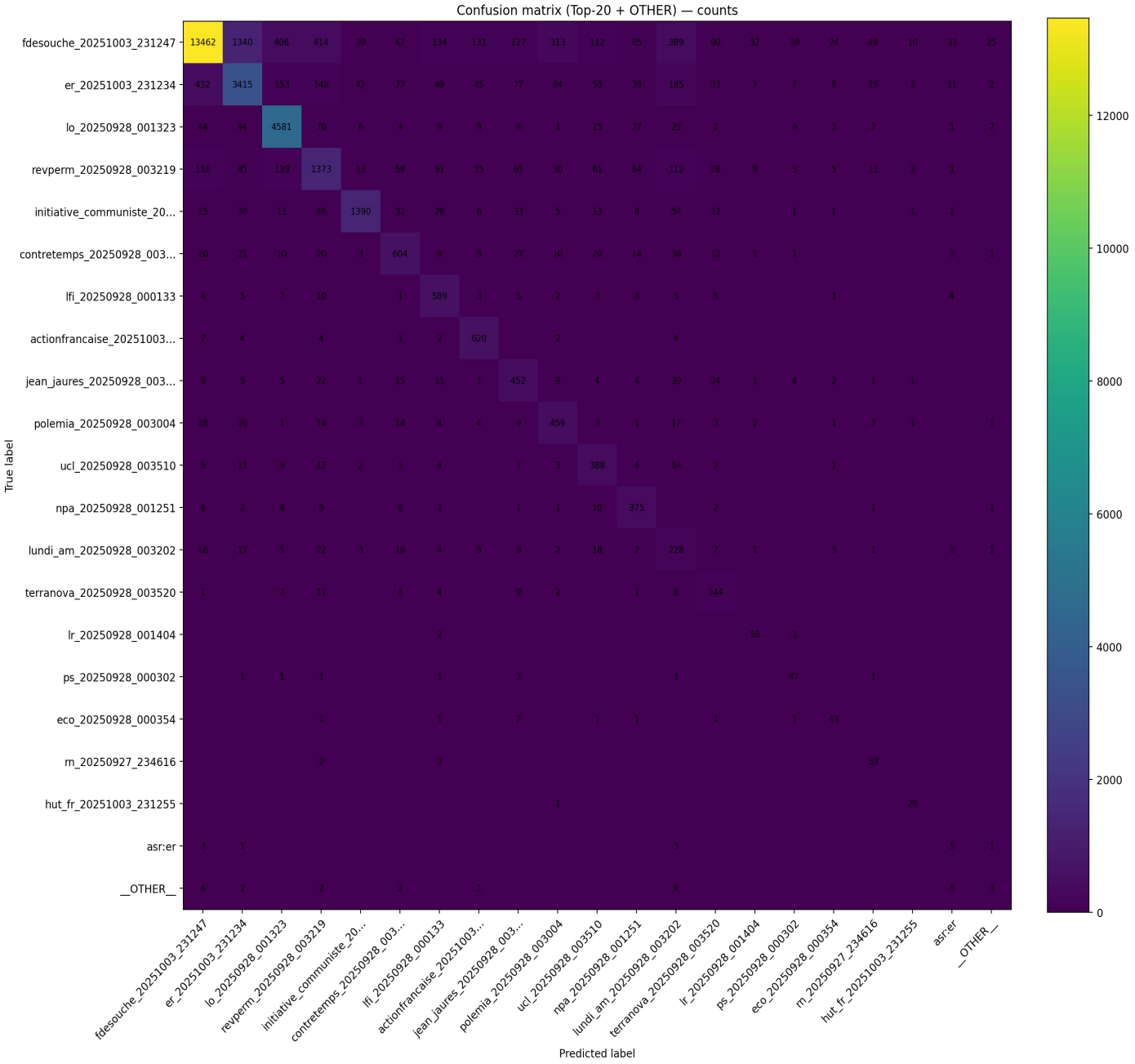
Lecture :

- Une **accuracy** relativement élevée peut coexister avec une **macro-F1 faible** lorsque certaines classes dominent et que les classes rares sont mal prédites.
- Les figures d’agrégation (groupes et top confusions) rendent ces effets immédiatement visibles.

Figures associées :

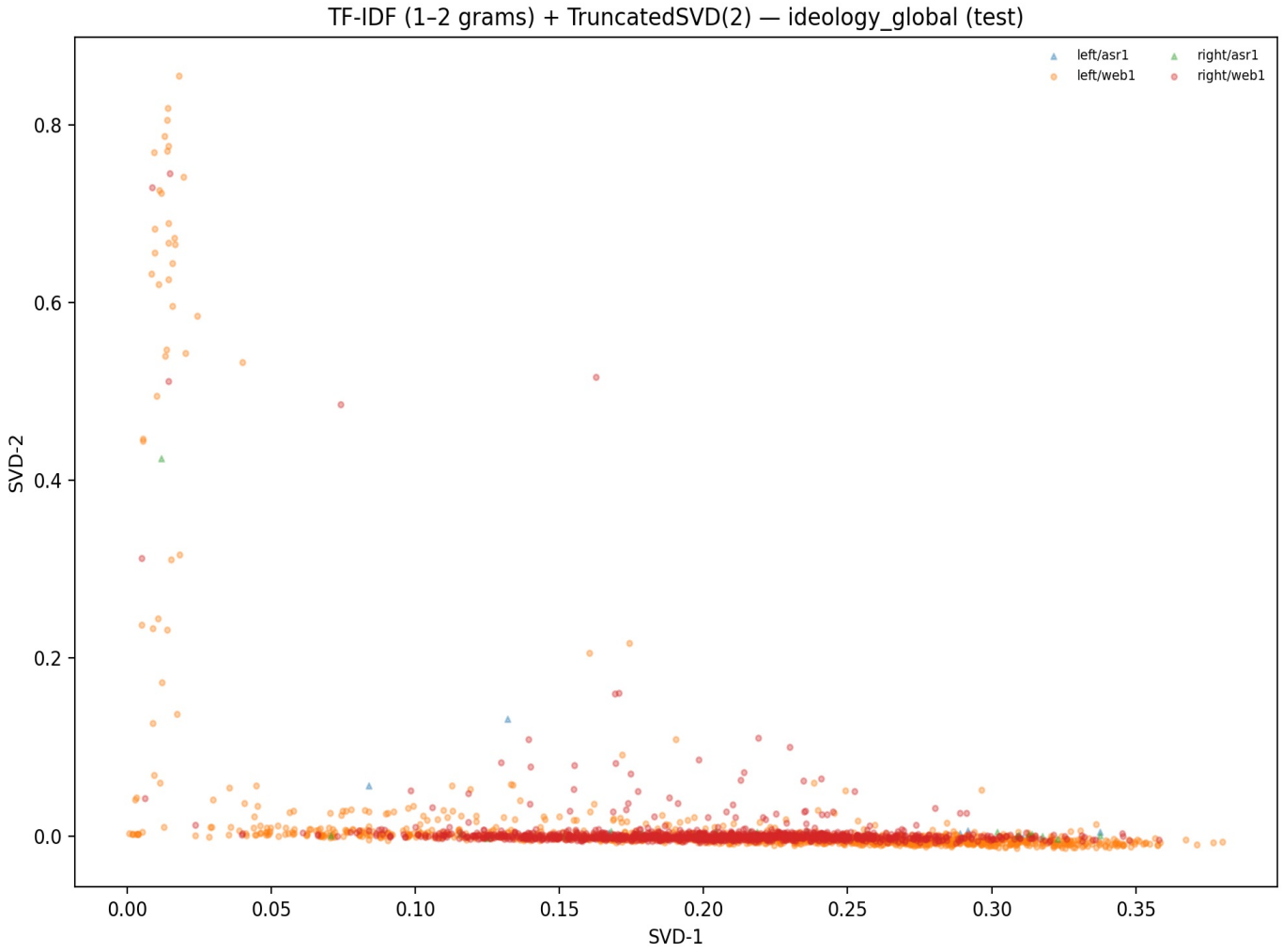

Confusion matrix — aggregated (ASR vs CRAWL)
ASR
CRAWL
OTHER
True group
14
19
84
35867
ASR
CRAWL
OTHER
Predicted group
35000
30000
25000
20000
15000
10000
5000
0


Confusion matrix (Top-20 + OTHER) — row-normalized
True label
Predicted label
fdesouche_20251003_231247
er_20251003_231234
lo_20250928_001323
revperm_20250928_003219
initiative_communiste_20...
contretemps_20250928_003...
lfi_20250928_000133
actionfrancaise_20251003...
jean_jaures_20250928_003...
polemia_20250928_003004
ucl_20250928_003510
npa_20250928_001251
lundi_am_20250928_003202
terranova_20250928_003520
lr_20250928_001404
ps_20250928_000302
eco_20250928_000354
rn_20250927_234616
hut_fr_20251003_231255
asr:er
__OTHER__
0.0
0.2
0.4
0.6
0.8



5.1.3 Projection SVD (exploration) - mix web1+asr1

La figure suivante fournit une visualisation exploratoire (TF-IDF + réduction SVD) et aide à interpréter une partie du domain shift :



5.2 Synthèse comparative (binaire vs multi-classe)

Tâche	Corpus	Modèle / run	n_test	Accuracy	Macro-F1	Commentaire
Ideology_global (binaire)	web1	`FINAL_web1_ideo` / `tfidf_svm_quick`	24781	0.972	0.967	Baseline forte (écrit)
Ideology_global (binaire)	asr1	`diag_asr1_ideo_base` / `tfidf_svm_quick`	94	0.755	0.745	Diagnostic (support faible)
Ideology_global (binaire)	mix web1+asr1	`web1_asr1` / `tfidf_svm_quick` (global)	37216	0.972	0.967	Dominé par web1
Ideology_global (balanced)	mix web1+asr1	`web1_asr1` / balanced_by_corpus (0.5/0.5)	-	0.846	0.788	Mesure robuste du domain shift
Crawl (~20 classes)	web1	`FINAL_web1_crawl` / **tfidf_smo_rbf**	51364	0.755	0.644	Meilleure macro-F1 (multi-classes)

Lecture méthodologique

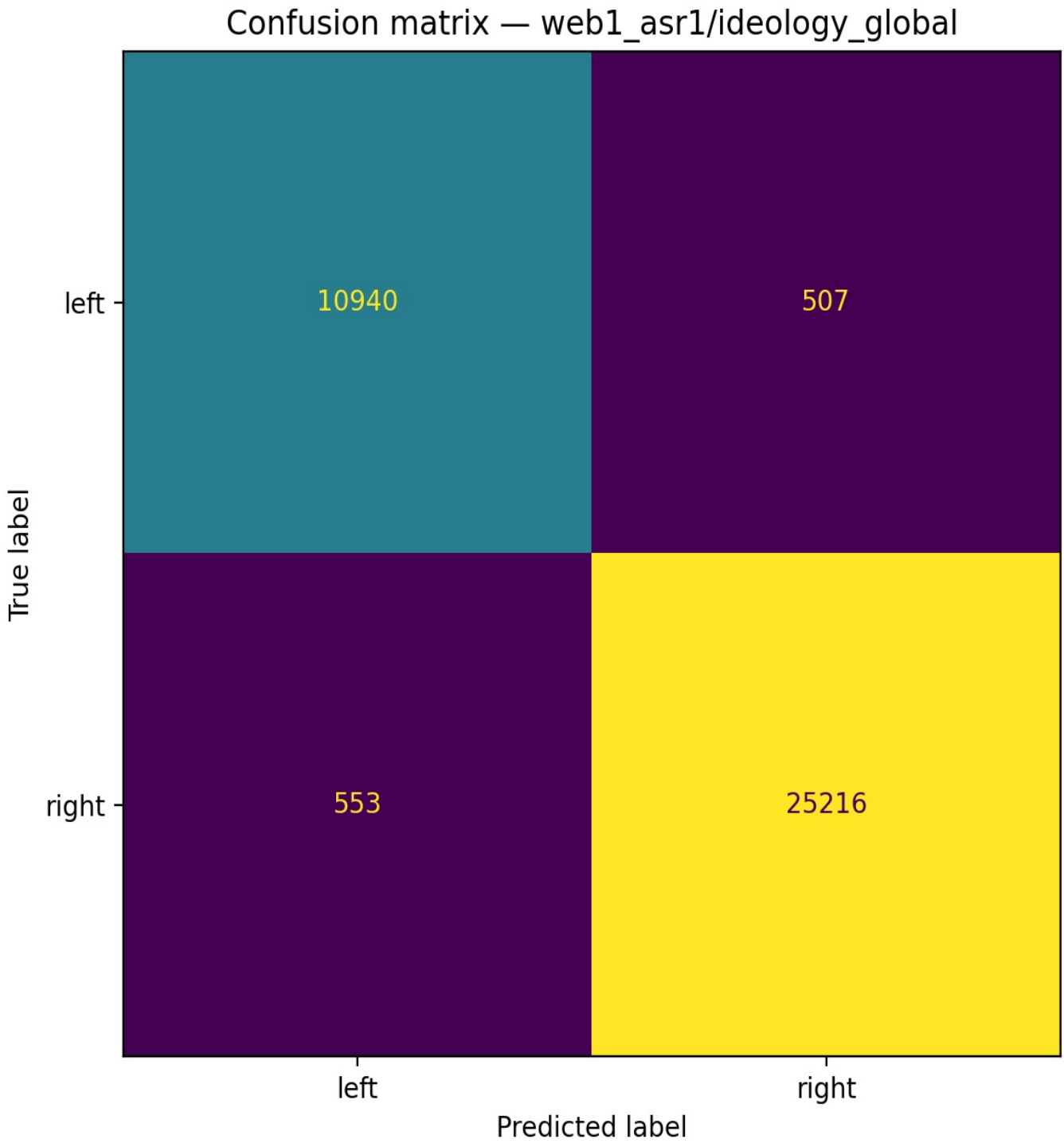
- **WEB-only** fournit la baseline la plus stable : gros volume, texte “propre”, et labels idéologiques globalement cohérents au niveau des acteurs.
- **ASR-only** matérialise le coût du bruit (ASR) et de la réduction de support après filtrage ; l’enjeu n’est pas d’“additionner” ASR à WEB, mais de définir un **protocole ASR** (filtrage, segmentation, modèle) pertinent.
- **Crawl multi-classe** est plus exigeant (20 classes + déséquilibres) : il sert davantage à évaluer la capacité du pipeline à produire des analyses **plus fines** que la simple binarisation, au prix d’un protocole d’équilibrage (ex. `cap_docs`) et d’une interprétation plus prudente des classes minoritaires.

5.3 Analyse qualitative

5.3.1 Analyse d'erreurs (Ideology)

L'objectif ici n'est pas de "sur-interpréter" les scores, mais de comprendre **où le modèle se trompe** et **ce que cela révèle** sur le signal appris.

Sur le run *mix* (`web1_asr1`, n_test = 37216), l'accuracy globale (0.972) correspond à environ **1060 erreurs** sur le jeu de test. Cette valeur est cohérente avec la matrice de confusion globale (Figure ci-dessous).



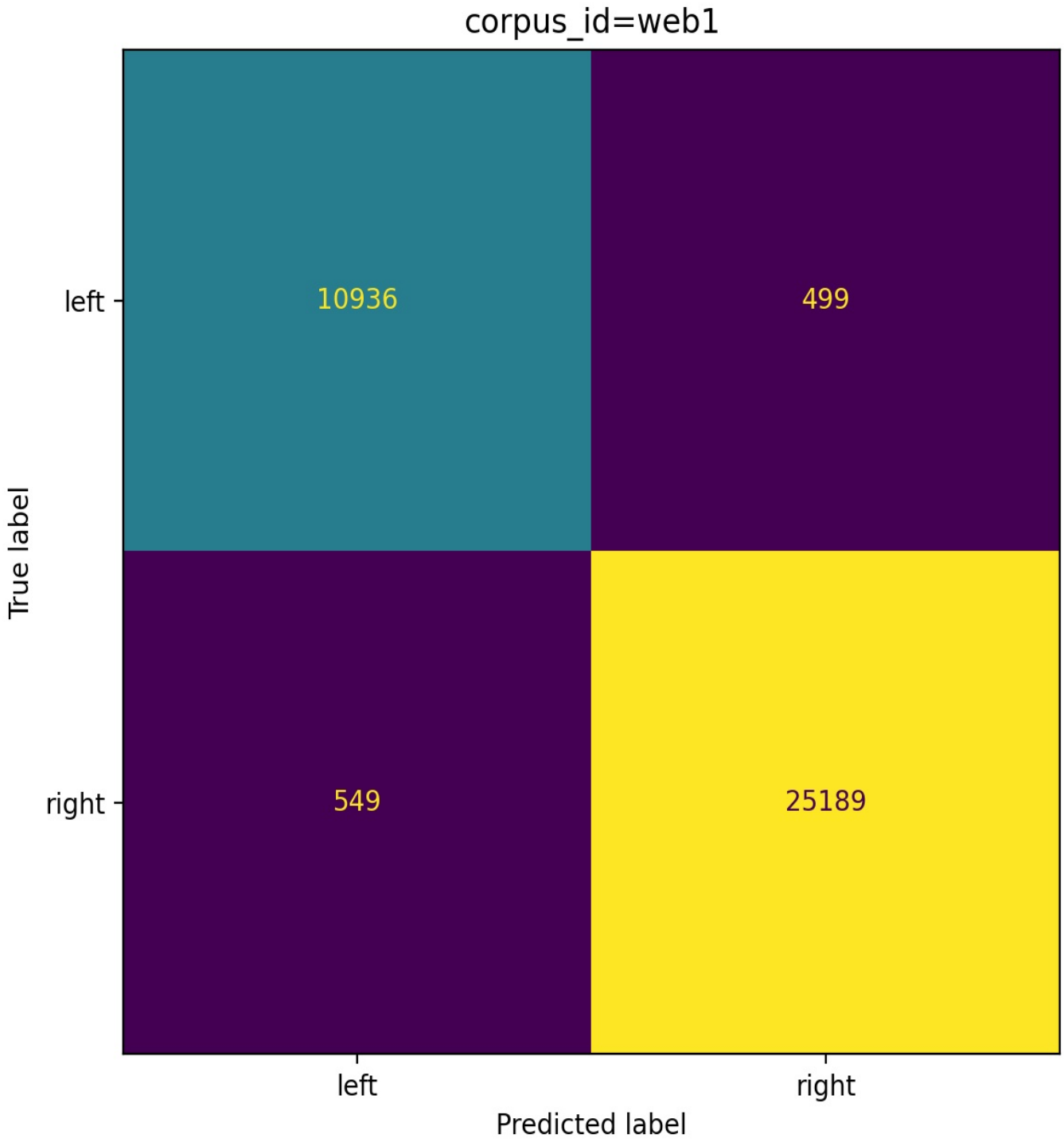
Points d'attention méthodologiques :

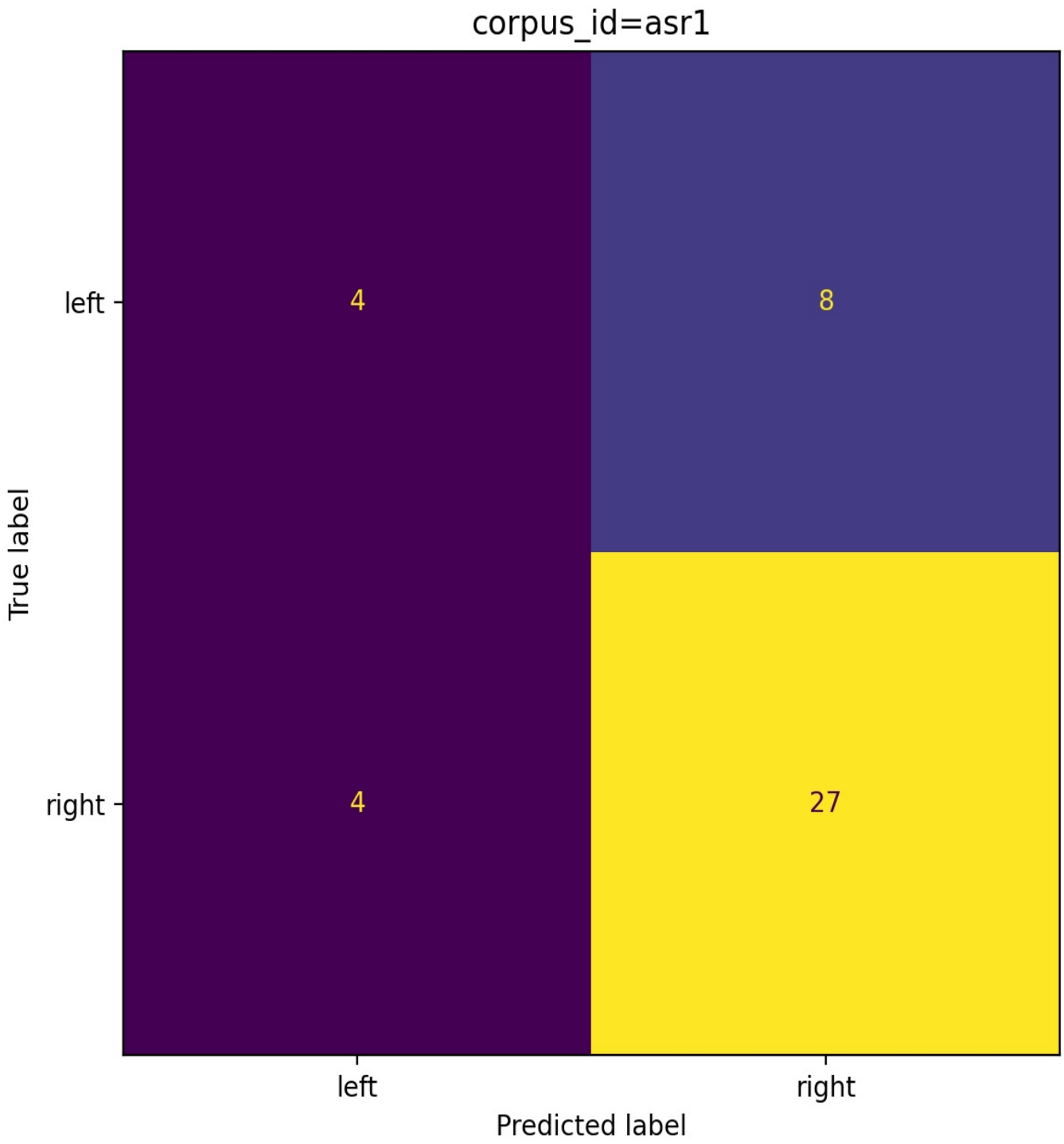
- Le support **ASR1** dans l'évaluation mix est faible (n=43), ce qui rend toute analyse fine instable sur ce sous-corpus.
- Pour une analyse qualitative rigoureuse, il faut échantillonner des erreurs et les annoter (thème, entités, cadrage, bruit ASR, etc.). Les fichiers `reports/*/meta_eval.json` + les splits TSV permettent de reconstruire un protocole d'échantillonnage sans ambiguïté.

5.3.2 Interprétation : ce que les figures révèlent (domain shift + signal composite)

Les figures “par corpus” montrent un contraste net :

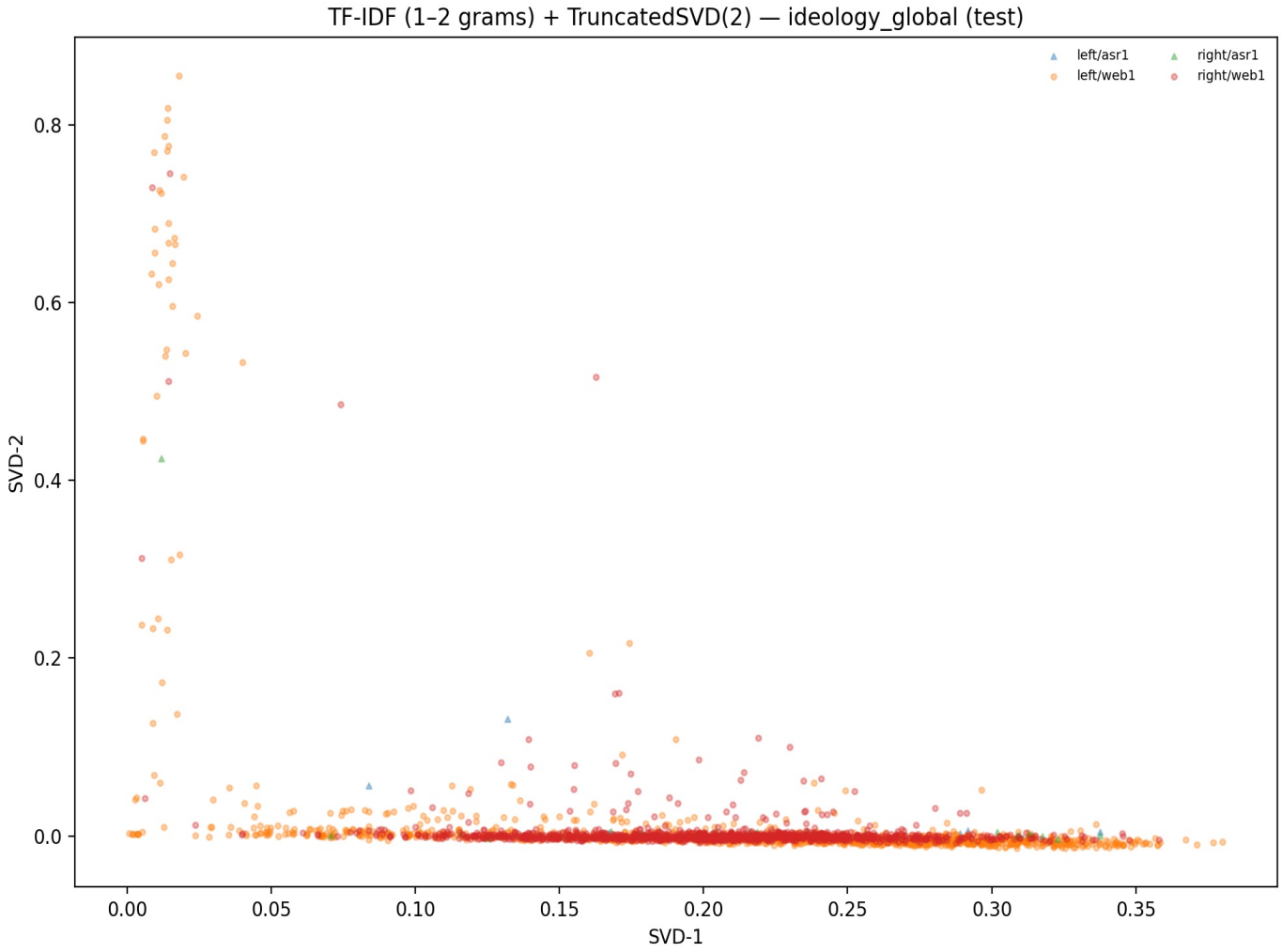
- **web1** : performances élevées et proches du run web-only ; le modèle exploite un signal riche (lexique + style + provenance).
- **asr1** : baisse substantielle (macro-F1 ≈ 0.609), compatible avec (i) bruit ASR, (ii) oralité, (iii) distribution thématique/ structurelle différente, et (iv) effet de déséquilibre (peu d’exemples ASR dans l’évaluation).





Au niveau “macro”, les métriques **balanced par corpus** (0.788) résument bien la situation : la performance “réelle” en comparaison web vs ASR est inférieure à la performance globale, car le web (très majoritaire) masque une partie des difficultés ASR.

Enfin, la projection SVD (exploration) suggère un **décalage distributionnel** entre les documents web et les transcriptions ASR, cohérent avec l’hypothèse “signal composite” et la baisse de robustesse hors domaine.



6. Conclusion et perspectives

6.1 Bilan

PEPM aboutit à un pipeline complet et reproductible, appliqué à des données originales (web + ASR). Le cœur du travail est méthodologique : rendre possible des expériences robustes sur gros volumes, et produire des artefacts et audits permettant une discussion critique des performances.

6.2 Limites

- contraintes CPU (pas de gros runs Transformers),
- déséquilibre web vs ASR,
- conversion brut -> TEI pragmatique (métadonnées riches en partie hors TEI),
- complexité d'un core très générique.

6.3 Perspectives

- tiny-run Transformers sur sous-échantillon,
- holdout strict par source/domain,
- analyses non supervisées (clustering thématique) adaptées au sujet initial,
- enrichissement TEI (réinjection contrôlée de métadonnées).

7. Annexes

A. Commandes de reproductibilité

A.1 Extraction des métriques FINAL

```
python -m venv .venv
source .venv/bin/activate
pip install -U pip
pip install -r requirements.txt
make setup
make pipeline PROFILE=ideo_quick
make pipeline PROFILE=ideo_quick_web1_asr1 DATASET_ID=web1_asr1
```

C. Bibliographie

Bibliographie (format simple, académique)

- **Barbaresi, A.** (2021). *Trafilatura: A Web Scraping Library and Command-Line Tool for Text Discovery and Extraction*. ACL (System Demonstrations).
- **Fan, L.**, et al. (2019). *In Plain Sight: Media Bias Through the Lens of Factual Reporting*. EMNLP-IJCNLP.
- **Kiesel, J.**, et al. (2019). *SemEval-2019 Task 4: Hyperpartisan News Detection*. SemEval.
- **Gururangan, S.**, et al. (2020). *Don't Stop Pretraining: Adapt Language Models to Domains and Tasks*. ACL.
- **Martin, L.**, et al. (2020). *CamemBERT: a Tasty French Language Model*. ACL.
- **Radford, A.**, et al. (2022). *Robust Speech Recognition via Large-Scale Weak Supervision*. arXiv (Whisper).
- **Laver, M.**, **Benoit, K.**, & **Garry, J.** (2003). *Extracting policy positions from political texts using words as data*. *American Political Science Review*.
- **Slapin, J. B.**, & **Proksch, S.-O.** (2008). *A scaling model for estimating time-series party positions from texts*. *American Journal of Political Science*.
- **Gentzkow, M.**, & **Shapiro, J. M.** (2010). *What drives media slant? Evidence from U.S. daily newspapers*. *Econometrica*.
- **TEI Consortium.** (2025). *TEI P5: Guidelines for Electronic Text Encoding and Interchange* (P5).