Threat-Modelled Design of a Trustworthy Multi-Model Cyberbullying Detection Service

Sumayyah Sophie Nandom
Department of Computer Science
Federal University Wukari, Nigeria
sumayyah@fuwukari.edu.ng

Suleiman Bawa Pyiki
Department of Software Engineering
Federal University Wukari, Nigeria
suleiman.bawa@fuwukari.edu.ng

Nandom James Milaham
Department of Software Engineering
Federal University Wukari, Nigeria
nandom.milaham@fuwukari.edu.ng

***Abstract*— Automated cyberbullying detectors are used as moderation systems but are rarely identified as systems of security concern. A threat-modeled design of a multi-model detection service is presented, and evidence previously claimed for an earlier prototype is discussed in the context of the detection claims. The design integrates six model families, per-model calibration, an agreement score, split-conformal prediction sets, and an abstain-to-human policy. It maps eight threats (evasion, spoofing, model extraction, poisoning, privacy leakage, artifact tampering, resource exhaustion, and weaponized false positives) to twelve testable security requirements. A recomputation of the reported prototype results, based on analysis of the confusion matrices and ROC curves, reveals that its three artifacts are inconsistent with each other. The accuracy range based on the confusion matrices (38.4% to 50.0%) is below that of an always-negative classifier (61.3%), and the AUC range (0.470 to 0.525) is indistinguishable from 0.5 (two-sided p from 0.11 to 0.96). A synthetic validation study with 100 replications reveals that conformal coverage is 0.901 in the source and 0.712 in the source shift, and that averaging calibrated probabilities does not outperform the best single learner when the learners' errors are correlated. An evaluation protocol with pre-specified statistics, behavioral tests, and evasion tests is provided. The empirical assessment of the design on real abusive-language data is not presented here and is the next step of the work.**

Keywords—Cyberbullying detection, threat modeling, adversarial evasion, conformal prediction, selective classification, content moderation, reproducibility.

I. Introduction

Cyberbullying negatively affects its victims, causing anxiety, depression, and, in extreme cases, suicidality [1]. Platforms and institutions use automated text classifiers to flag abusive messages for deletion or review, and regulatory and risk-management systems require providers to implement them with documented safeguards [2], [3]. When used in this way, a detector is classified as a security control [4]. It makes decisions that can be triggered by adversaries, sidesteps or exploits them, and processes personal text that needs to be safeguarded [5].

There is a lack of uniformity in the literature on security abuse detectors: classifiers are brittle. Small character-level modifications reduced the toxicity score for a production API [6], and typographical changes and word-boundary changes, plus innocuous word insertions, foiled seven published hate-speech models [7], and recent tests of cyberbullying detectors indicate up to 88% and 44% attack success rates at the word-level and character-level, respectively [8]. Additional attack surfaces include unrestricted output of model scores for extraction [9], user feedback for data poisoning [10], and stored submissions for privacy exposure [11], [12] via the service interface.

The issue is worsened by evaluation practice. The field has been criticized for its lack of consensus on the definition of cyberbullying, changing assessment procedures, and unrepeatable reporting [13], [14]. When training and test items come from the same sources and/or are written by the same authors, the within-dataset validation overestimates the performance. Most published designs optimize only one or two sets of models within a single model family and don't specify what the system should do if its models disagree [15] or if their probabilities aren't reliable [16].

Both gaps are depicted in the work of Balogun [17]. It combines six classifiers (SVM, Naive Bayes, CNN, LSTM, BERT, and RoBERTa) and exposes them through a web interface, showing the output of each model for each input. The prototype claims superiority of the transformer, but this is not reflected in the confusion matrices and ROC curves, and the quantities in the interface are not explained [18]. However, the prototype is a good starting point because it uncovers the mismatch between the model families, which can then be turned into an explicit uncertainty signal if the design is principled.

This paper has four contributions. First, it defines a threat model for a multi-model abuse-detection service and 12 security requirements, each associated with a security control and a verification test. Second, it defines a service-oriented architecture in which per-model calibration, an agreement score, and split-conformal prediction sets are used to implement an explicit abstain-to-human policy, along with the class, sequence, and interface designs that realize it. Third, it audits the evidence reported for the prototype by recomputing the metrics with confidence intervals and significance tests and validates the abstention mechanism with synthetic data with a known structure. Fourth, it provides an evaluation protocol (statistics and test suites) that a follow-on study can run on the public data set utilized by

the prototype. The paper is organized around four research questions:

- RQ1 - What are the threats and security requirements for a multi-model cyberbullying detection service and what are the design controls that address them?
- RQ2 - Does the evidence reported for the prototype support its detection claims?
- RQ3 - How can calibrated agreement and conformal abstention be formulated to make the error control explicit, and how do they perform in a source and in case of source shift?
- RQ4 - What evaluation protocol should be followed to determine the detection performance, evasion robustness and behavioural validity of such a service?

The design (Sections III and IV) addresses RQ1, the protocol (Section V) addresses RQ4, and the analyses (Section VI) address RQ2 and RQ3. There is no new evaluation of the six models on real data, and no claim is made about the relative quality of the models on abusive-language data. The related work is summarized in Section II; findings are discussed in Section VII; threats to validity are addressed in Section VIII; ethics and privacy are covered in Section IX; and the conclusion is given in Section X.

II. Background and Related Work

A. The construct and the data

Cyberbullying is defined as "aggression that is intentional, repeated, and directed against a victim who cannot defend him or herself in a situation where he or she feels harmed and is likely to continue to feel harmed in the future, and that occurs through an electronic medium [19]. " Automated detection usually focuses on individual messages and thus would detect aggression but not repetition or power imbalance. The difference between bullying roles and types in annotation schemes reveals the extent to which a message-level label lacks context. In a systematic review of 22 studies, it was found that the literature on cyberbullying is sometimes misleading, and methodology is not consistent in evaluating it [13].

Publicly available data on English cyberbullying detection is heterogeneous. The prototype's aggregate comprises subsets from multiple platforms and tasks: Wikipedia talk-page aggression, attack, and toxicity annotations [20]; data from Twitter on racism and sexism; a set from YouTube; and a Kaggle set of insults [21]. Annotations for repetition, peerness, and intent to harm are added synthetically to a later dataset, also derived from the same text messages [22]. Label definitions are therefore different from one source to another, and all the evaluation designs mentioned below are relevant.

B. Detection approaches and recent advances

Previous systems employed classical learners or lexical rules rather than bag-of-words or TF-IDF features, and comparisons between SVM and Naive Bayes classifiers remain prevalent [23]. Deep models include learned embeddings and convolutional and recurrent encoders: one comparison on Twitter hate data employed n-gram ranges to compare machine-learning and deep-learning classifiers. Pretrained transformers have made a significant leap forward: a fine-tuned BERT model outperformed previous cyberbullying detectors [24], and a study of large language models for the task identified RoBERTa as one of the best models [25]. These trends are summarized in surveys [26], and a recent review states that large language models achieve the best performance but consume high energy and are not used in real time.

Evidence from 2024 and 2025 expands the evidence base. In the case of YouTube comments, a benchmark of large language models is presented that explores prompting strategies for cyberbullying detection [27], while a comparison of text-classification methods across different datasets assesses their accuracy and computational costs [28]. Offensive language datasets and transfer approaches are reviewed systematically, and an adversarial encoding framework is used to adapt a transformer to a platform other than the one on which it was trained [29]. These studies consider generalisation a performance issue. They don't consider the detector to be a service with a threat model.

C. Evaluation validity

Cyberbullying and hate-speech detectors are tested on datasets with strong source and author effects. Models trained on one data type and tested on another perform worse than models trained to ignore these effects and tested on multiple types [7], or they overestimate performance [30]. The main drawbacks of the field, identified in a critical survey, are evaluation criteria, data scarcity, and reproducibility [14]. These findings drive source-held-out evaluation, trivial-baseline reporting, and complete configuration disclosure in the protocol of Section V.

D. Adversarial evasion and the security of abuse detectors

Misspellings, punctuation insertions, and splitting words to evade toxicity classifiers were demonstrated against a deployed API, along with false-alarm attacks that make benign phrases appear toxic [6]. The most successful attacks against seven hate-speech models [7] were either dilution with innocuous words or removal of word boundaries. Lexical perturbations are sensitive to cyberbullying classifiers, and augmenting classifiers with such perturbations provides robustness [31]. A 2025 study on the robustness of cyberbullying detectors against word substitution and character perturbation attacks in text showed that language-model-based detectors were more resilient to both types of attacks than other models tested, and that adversarial training increased their robustness [8]. The tests become routine using general attack toolkits. In addition to evasion, data poisoning can introduce hidden behavior in NLP models [10], prediction APIs can leak model parameters [9], and trained models can leak information about training records [12]. Within guard models for conversational

systems, the safety classifiers are considered as a component with their own risk profile.

E. Uncertainty, calibration, and abstention

Modern neural classifiers tend to be poorly calibrated, and post hoc scaling decreases the calibration error [32]. Platt scaling maps the classifiers' scores to probabilities [33], and ECE summarizes the difference between confidence and accuracy [34]. Selective classification allows a classifier to refuse a classification decision, thereby avoiding the risk of an incorrect classification on refused cases at the cost of increasing the risk on accepted cases [35]. Conformal prediction is a method that transforms any score into a set of predictions that has finite-sample marginal coverage under exchangeability [36]. The technical basis here is conformal prediction, which has been applied to toxic-content classification [37]. The coverage guarantee is not valid for distribution shift, which the design must monitor rather than assume is valid.

F. Behavioural testing and fairness

Behavioural testing tests models with small, controlled inputs and shows failures that are otherwise masked by accumulated accuracy [38]. A good test suite for hate-speech classifiers includes negation, reclaimed slurs, and counter-speech [39]. However, data on abusive language is racially biased, and that bias is reflected in models [40]; hate classifiers misclassify dialect and use of slurs as abusive language [40]. These results validate the identity-term and benign-profanity test cases of the test suite in Section V.

G. Governance and documentation

Model cards and dataset datasheets provide standardization in disclosure of intended use, evaluation conditions, and limitations [41]. It is crucial that AI systems are measured and governed within risk-management frameworks [3]. The submitted text is protected by data protection regimes [42]–[44]. The design takes these instruments as deliverables (requirement SR12 in Section III).

H. Positioning

Table I lists the most closely related studies to the present design. None of the studies in the table explicitly define a detection service with a threat model, a calibrated uncertainty, and an abstention policy, and none considers abstention in the context of source shift on a multi-source cyberbullying corpus.

TABLE I. CLOSELY RELATED WORK AND ITS CONNECTION TO THE PROPOSED DESIGN. RELATION: HOW THE CITED WORK RELATES TO THE DESIGN OR WHAT IT LEAVES OPEN

Study	Contribution	Relation to the proposed design
Rosa et al. [13]	Systematic review of 22 studies plus an experiment on two datasets	Motivates explicit construct definition and a uniform evaluation protocol
Emmery et al. [14]	Critical survey of evaluation criteria, reproducibility and data scarcity	Source of the reproducibility and baseline requirements in Section V
Arango et al. [30]	Shows overestimation from within-dataset validation of hate-speech models	Motivates grouped and source-held-out splits
Hosseini et al. [6]	Evasion and false-alarm attacks on a toxicity API	Defines threats T1 and T8; no defence is specified
Gröndahl et al. [7]	Seven models brittle to typos, word-boundary changes and dilution; data type matters more than architecture	Defines the evasion operators of the test suite
Juarez et al. [8]	Word- and character-level attacks and countermeasures for cyberbullying detectors	Provides attack baselines; does not address uncertainty or abstention
Yi and Zubiaga [29]	Adversarial encoding for cross-platform transfer	Addresses shift by adaptation; the present design measures and monitors it
Villate-Castillo et al. [37]	Conformal prediction for toxic-content classifiers	Supplies the conformal mechanism, which is evaluated here under source shift
Muminovic [27]	Benchmark of large language models on YouTube comments	Candidate baseline family for the protocol
Philipo et al. [28]	Comparison of text-classification methods across datasets with attention to cost	Closest comparison study; informs the cost metrics of Section 5
Ogunleye and Dharmaraj [25]	Language-model evaluation for cyberbullying detection	Reference for transformer baselines
Prototype [17]	Six-model web system with per-model display	Starting point; audited in Section VI.A

III. THREAT MODEL AND SECURITY REQUIREMENTS

A. Assets, trust boundaries, and assumptions

The service ensures the security of six assets: the integrity of its decisions; its model artifacts; training data and feedback; submitted text and audit records; its availability; and the capacity of its human reviewers. The data flows for this situation are illustrated in Fig. 1, together with four trust domains. The client zone (TB1) is not trusted. The application tier (TB2) handles the decryption of TLS and validates the input and caller. Inference workers, calibrators, and the model registry are all hosted by the model tier (TB3). The data tier (TB4) stores the audit log, the review queue, and training data. The design relies on a trusted identity provider, as well as trusted, isolated containers for the model workers and authenticated human reviewers.

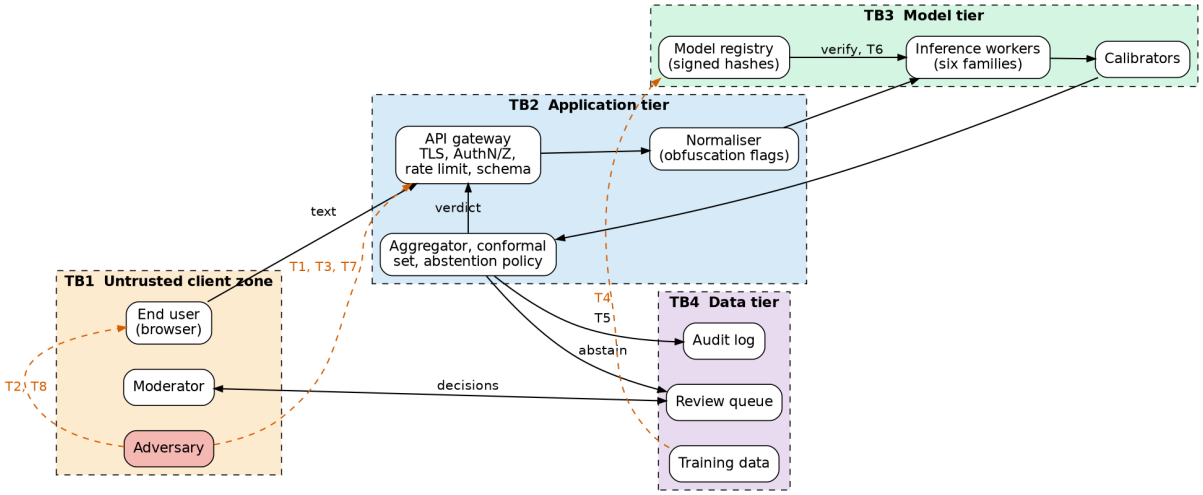

Fig. 1. Data-flow diagram of the service with trust boundaries TB1, TB2, TB3, and TB4. Adversary-controlled flows are indicated by dashed orange edges and the threat identifiers of Table II. The solid edges represent valid data flow. Edge labels indicate the data or threat associated with the flow.

B. Adversaries and threat catalogue

Five adversary roles are considered; An evasion adversary can make black-box queries and modify the abusive text until it is accepted. A poisoning adversary can provide feedback or training examples. A weaponizer is someone who creates benign-looking text and then gets the system to flag it or overload the review queue. The extractor makes numerous queries to determine a model. A submission is read by an insider or curious operator. Eight threats are identified that correspond to the identified roles and are listed in Table II, along with their STRIDE and LINDDUN categories, the asset they affect, and the supporting evidence.

TABLE II. THREAT CATALOGUE. THE CATEGORIES ARE BASED ON STRIDE (S SPOOFING, T TAMPERING, I INFORMATION DISCLOSURE, D DENIAL OF SERVICE) AND LINDDUN (L LINKABILITY, I IDENTIFIABILITY, D DISCLOSURE). EVIDENCE: PUBLISHED DEMONSTRATIONS OF THE THREAT CLASS

ID	Threat	Category	Asset	Evidence
T1	Evasion by character, word-boundary or dilution edits that lower the abuse score	T (decision integrity)	Decision integrity	[6] – [8], [31]
T2	Credential abuse or token theft at the gateway; calls without authorisation	S	Availability, data	Design analysis
T3	Model extraction through unrestricted per-model scores	I	Model artefacts	[9]
T4	Poisoning through feedback or contributed training data	T	Training data	[10]
T5	Privacy exposure through stored submissions, history and logs	L, I, D; I	Submitted text	[11], [12]
T6	Tampering with model files or the registry manifest	T	Model artefacts	Design analysis
T7	Resource exhaustion through long inputs or request bursts	D	Availability	Design analysis
T8	Weaponised false positives: benign text engineered to be flagged, or review-queue flooding	T, D	Decision integrity, reviewer capacity	[6]

C. Security requirements

The threat catalogue results in 12 requirements. Each is expressed in a way that can be tested for by a test defined in Section V and is associated with Table III which shows the threats it addresses.

TABLE III. SECURITY REQUIREMENTS, THE THREATS ADDRESSED AND THE VERIFICATION METHOD. THE E-CODES ARE THE PHASES IN THE EVALUATION PROTOCOL IN FIG. 9.

ID	Requirement	Threats	Verification
SR1	Every endpoint requires authentication and role-based authorisation with short-lived tokens	T2	Authorisation test matrix over all roles and endpoints (E7)
SR2	Rate limits and quotas apply per identity and per address	T2, T3, T7	Burst test; measure rejected-request rate and latency under load (E7)
SR3	Inputs are validated against a schema, with a hard length limit of 500 characters and Unicode normalisation	T1, T7	Fuzzing with malformed, oversized and mixed-script inputs (E7)
SR4	Normalisation is obfuscation-aware and flags every text it modifies	T1	Evasion suite: flag recall per operator (E6)
SR5	External responses expose label, prediction set and status; per-model scores are restricted to authorised roles	T3	Response-schema test; query-budget extraction experiment (E7)
SR6	Model artefacts are loaded only after signature and hash verification against a signed manifest	T6	Tamper test: modified artefact must fail to load (E7)

SR7	Training data carries a provenance record; duplicates are removed; feedback is not used for retraining without review	T4	Provenance audit and poisoning canary test (E0, E7)
SR8	Raw text is not retained by default; audit events are pseudonymous with a retention limit; exports are redacted	T5	Retention test; log inspection for raw text (E7)
SR9	Every decision reports calibrated probabilities	T1, T8	ECE, Brier score and NLL with CI (E3)
SR10	Non-singleton conformal sets are routed to human review	T1, T8	Coverage and abstention rate within source and under shift (E5)
SR11	Behavioural and evasion suites run before release and after each model update	T1, T8	Failure and attack-success rates with Wilson CI (E6)
SR12	Model card, dataset datasheet and change log are published for each release	All	Documentation checklist (E9)

IV. Design of the Detection Service

A. Architecture

The four-layer architecture is shown in Fig. 2. The presentation layer is a single page app implemented using React. The gateway layer terminates TLS, validates tokens issued by an external identity provider, and applies rate limits and schema checks. The decision layer is a Flask service where the normalizer, model adapters, calibrators, aggregation, and policy components reside. The model registry, the pseudonymous audit log, the review queue, and the containerized offline training and evaluation environment are stored in the data layer. The layering positions all of the security controls defined in Table III at a specific border.

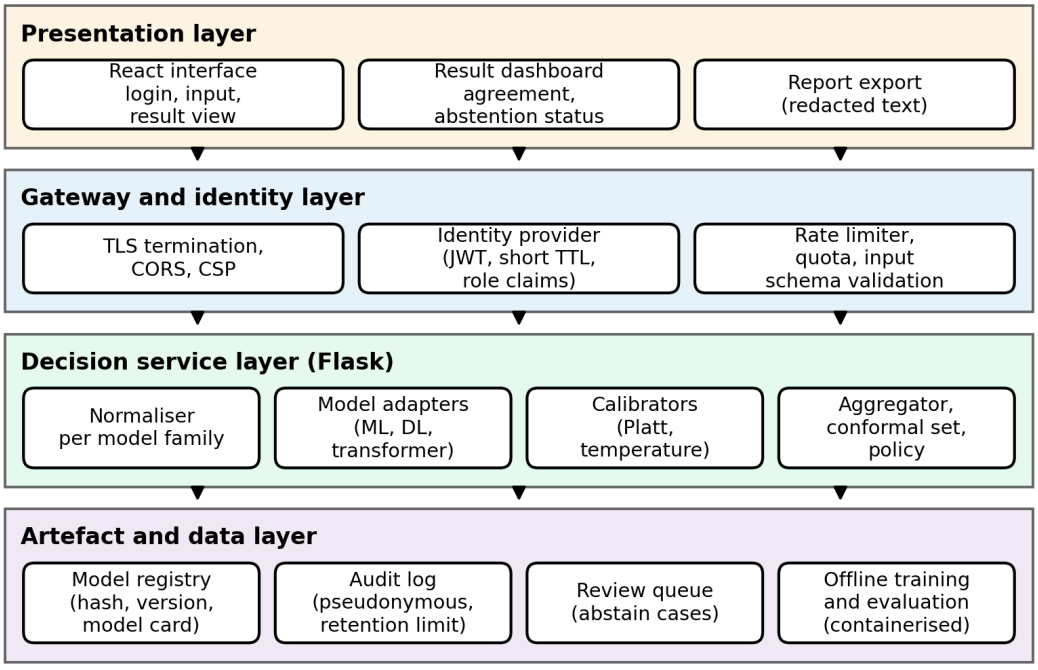

Fig. 2. Layered architecture of the proposed service. The layers (presentation, gateway, decision, data) are distinguished by color. Arrows show the direction of calls between layers

B. Inference pipeline and formal definitions

The inference pipeline is illustrated as shown in Fig. 3. The input is verified, normalized within each model family, and scored by six models (SVM and Naive Bayes with TF-IDF features, CNN and LSTM with word embeddings, and BERT and RoBERTa with pretrained checkpoints [45], [46]). The design change from the prototype (see report) is the per-family normalization, while stop-word removal and lemmatization are general pre-processing steps, and the text in the sample of the dataset is in lower case and stripped of punctuation. The raw text is given to the transformers since removing negations and function words changes the meaning. Notation is defined in Table IV.

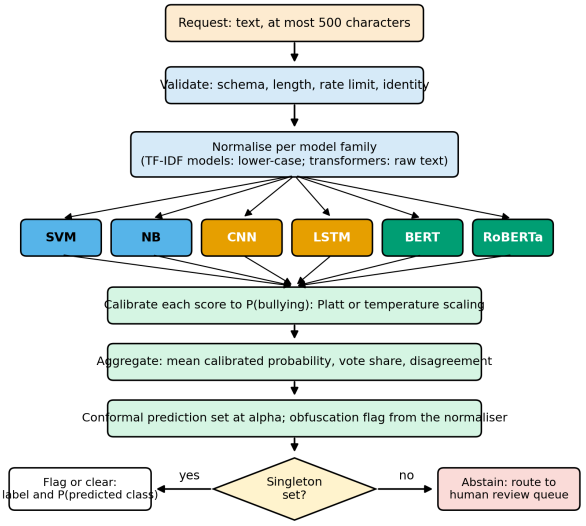

Fig. 3. Pipeline from Request to Verdict. The colours of model boxes indicate the families (blue: classical, orange: deep, green: transformer). The decision diamond is used to check if there is one label in the conformal set.

TABLE IV. DESIGNATION USED IN THE DESIGN.

Symbol	Meaning
x, y	Input text and label (1 = bullying, 0 = non-bullying)
M	Number of models (M = 6)
N_f(x)	Normalised text for model family f
s_m(x)	Raw score of model m
p_m(x)	Calibrated probability of the bullying class from model m
$\bar{p}(x)$, v(x), δ(x)	Mean calibrated probability, vote share and mean absolute disagreement
α, $\hat{q}$, Γ(x)	Target error rate, calibration quantile and conformal prediction set

Each score is calibrated on a held-out calibration split with Platt scaling for margin-based models [33] or temperature scaling for neural models [32]:

$$p_m(x) = \sigma(\, a_m \cdot s_m(N_f(x)) + b_m \,) \qquad (1)$$

The mean calibrated probability, the vote share and the disagreement are:

$$\bar{p}(x) = (1/M)\, \Sigma_m\, p_m(x), \quad v(x) = (1/M)\, \Sigma_m\, 1[p_m(x) > 0.5], \quad \delta(x) = (1/M)\, \Sigma_m\, |\, p_m(x) - \bar{p}(x)\, | \qquad (2)$$

The conformal layer uses the nonconformity score s(x, y) = 1 − $\bar{p}$_y(x), with $\bar{p}$_1 = $\bar{p}$ and $\bar{p}$_0 = 1 − $\bar{p}$. With the following n calibration pairs, threshold and prediction set are [36]:

$$\hat{q} = Quantile_\{\lceil (n+1)(1-\alpha) \rceil / n\}\ \{\, s(x_i, y_i)\, \}, \quad \Gamma(x) = \{\, y : s(x, y) \le \hat{q}\, \} \qquad (3)$$

The set is said to be marginally covered under exchangeability of the calibration and test items:

$$P(\, y \in \Gamma(x)\,) \ge 1 - \alpha \qquad (4)$$

The policy labels if |Γ(x)| = 1 and abstains otherwise. For $\hat{q}$ > 0.5 the abstention region is the interval 1 − $\hat{q}$ ≤ $\bar{p}$(x) ≤ $\hat{q}$. An empty set occurs when $\hat{q}$ < 0.5 and $\hat{q}$ < $\bar{p}$(x) < 1 − $\hat{q}$, and it is also routed to review. The obfuscation flag that is set by the normaliser is not reflected in the label. It is logged, and it can be set to a more stringent α when flagged inputs are used. The agreement indicates that v(x) and δ(x) are alternatives to $\bar{p}$(x) as the basis of the score and the relative value is assessed in Section VI.B and in protocol stage E5.

C. *Software design views*

These use cases are shown in Fig. 4. The service has four actors: end user, moderator, administrator, and security auditor. The prototype's use-case model included only the user and the administrator, and it reallocated training and preprocessing to a system actor, while the updated use-case model includes the review and verification functions that are needed by SR8 and SR10. The class design is in Fig. 5. Three implementation families are abstracted away by a ModelAdapter, and the artifact hash is checked before scoring (SR6). The order of the pipeline is as follows: Calibrator, Aggregator, and ConformalPolicy classes, and the application classes do not contain any credentials, as the identity is delegated to the identity provider. Fig. 6 illustrates the steps for a single analysis request from submission to abstention handling.

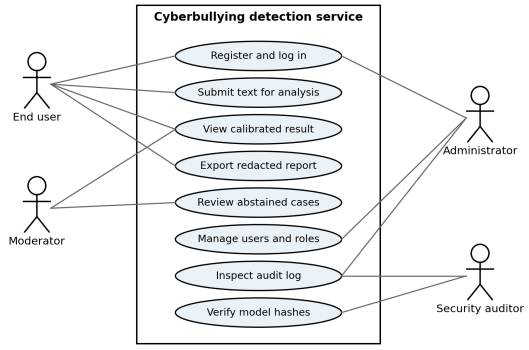

Fig. 4. Four actors use case diagram of the proposed service. The system boundary includes those functions that are visible to the user and those that are used to operate the system. Offline training is beyond the scope.

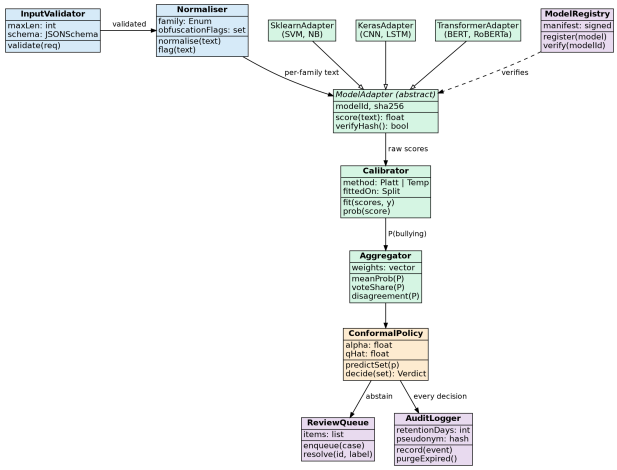

Fig. 5. Class diagram of the decision layer. Abstract ModelAdapter is marked with hollow arrowheads. Solid arrows show the call order of the inference pipeline. The registry verifies hash with a dashed arrow.

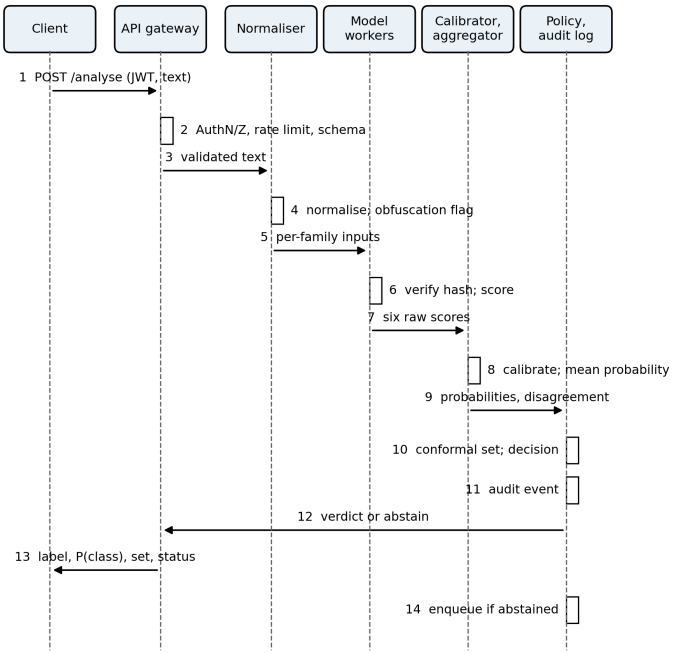

Fig. 6. Sequence diagram of one analysis request. Numbers are used to indicate the order of messages. The abstention is the only verdict in which the message 14 is raised.

D. *Interface design*

The baseline prototype [17] is shown in Fig. 7. It includes a login/sing-up page, a text input page (up to 500 characters), a result card displaying the output of each model, and a dashboard including a consensus chart and a model-agreement panel and a radar of harm indicators. The prototype is built using the following technologies: React, Tailwind CSS, Flask, and a hosted authentication service, and the models are built using scikit-learn, TensorFlow with Keras, and PyTorch. The screens for a benign input are reproduced in Appendix C.

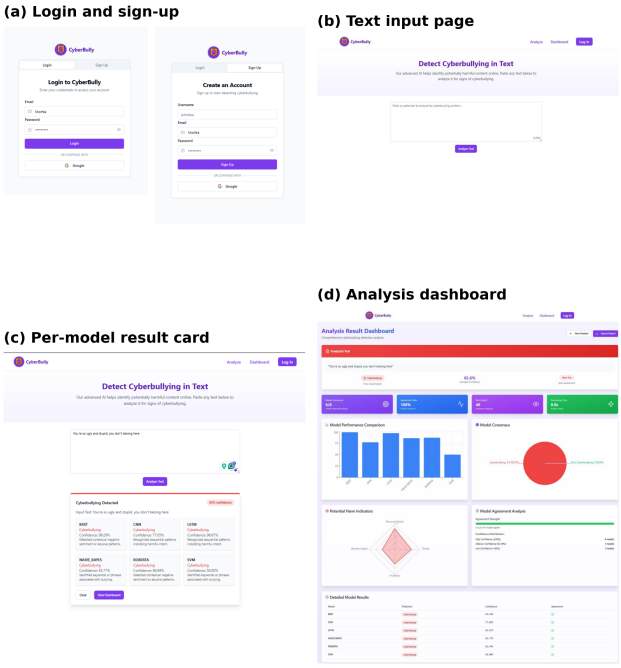

Fig. 7. The baseline prototype screens are (a) login and sign-up, (b) text input, (c) per-model result card (for an abusive example), and (d) analysis dashboard. The screenshots below have not been modified from the prototype.

The prototype interface inspection revealed four defects, which were corrected in the redesign. The confidence shown is the probability of the bullying class given the predicted class is non-bullying, and a benign text can be outputted with a 0.19% confidence. The SVM panel displays only 50.00% in two mutually contradictory predictions, meaning that it is not a calibrated probability but merely a placeholder. The model performance comparison chart plots per-input scores and not performance. There is no associated label, model, or validation for the harm-indicator radar. The redesign in Fig. 8 includes the calibrated P(bullying) per model, the conformal set, the verdict, an explicit abstention state, a normalized form when an obfuscation flag is raised, masking of displayed slurs, and that the text is not stored.

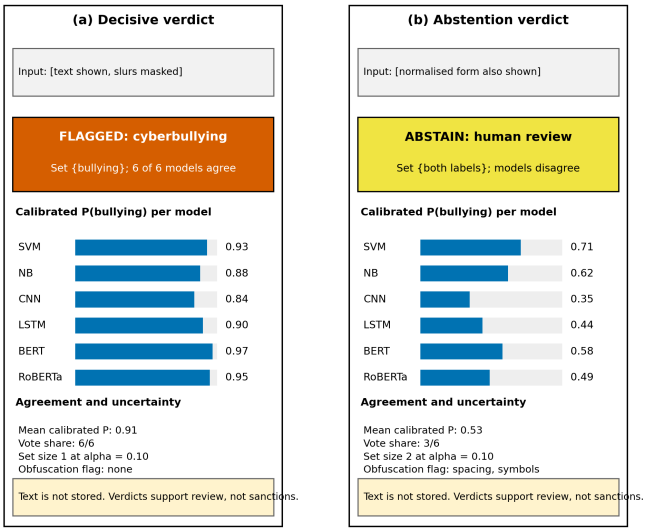

Fig. 8. Redesigned result view (wireframe - not yet implemented). In panel (a) there is a clear conclusion with 6 models in agreement. The abstention in part (b) is for disagreeing models and the raised obfuscation flag. The wireframe probabilities are indicative and not results.

E. Design verification by traceability

Every threat was traced to at least one requirement and every requirement was traced to at least one design element and one verification stage to verify the completeness of the design. There is a design element and a verification stage for each of the twelve requirements, and each of the eight threats has one or more requirements addressed to it. Traceability is used to provide completeness of the specification. It does not prove the controls are effective, which means that stages E3 to E7 must be executed.

> **Answer to RQ1.** There are eight threats (T1 to T8) that affect a multi-model cyberbullying detection service and twelve verifiable security requirements (SR1 to SR12). Each requirement is linked to a design element and a verification step, thus the specification is complete with regard to the threat catalogue. The protocol stages E3 to E7 are still required to demonstrate the effectiveness of the controls.

V. Evaluation Protocol and Analysis Methods

A. Overview

The ten stages of the protocol and the associated research questions are shown in Fig. 9. Any deviation from the protocol must be reported with the reason, and the protocol is specified before execution. The two analyses described in Section VI - the audit of the prototype's results and the synthetic validation of the abstention mechanism are also described in this section.

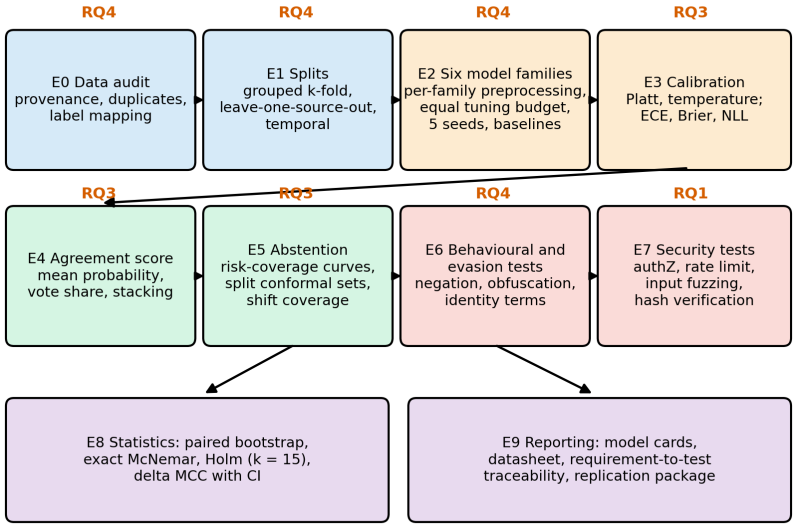

Fig. 9. Evaluation level stages E0 - E9. The colours blue, orange, green, red and purple represent data stages, model and calibration stages, agreement and abstention stages, validation stages, analysis and reporting stages, respectively. Each stage is identified by an orange label containing the research question that the stage serves.

B. Data, splits and models

The protocol is based on the public multi-source corpus that the prototype [21] has used. Stage E0 records for each source file, the platform, period, number of rows, positive share and original label definition, deletes exact and near duplicates with a specified threshold, and retains a source identifier for each item. Within each source, three split regimes are applied: grouped stratified five-fold validation (keeping duplicate clusters in the same group); leave-one-source-out (where a source is held out completely); and a temporal split (where timestamps exist). Augmentation is only done within training folds. Stage E2 is a training stage with the six model families and three baselines (majority class, stratified random and TF-IDF with logistic regression) and is trained with an equal tuning budget and five seeds for each model. Stage E3 fits calibrators on a validation split and reports expected calibration error with 15 bins [34], the Brier score, and the negative log-likelihood.

C. Metrics and statistical plan

The main metric is the Matthews correlation coefficient, which is informative under class imbalance [47], and macro-F1, PR-AUC, and ROC-AUC are secondary metrics. Proportions have Wilson intervals [48]. To assess the differences between models, the same test items are used for each model, and the McNemar test for accuracy is used as well as a paired bootstrap with 10,000 resamples for MCC and macro-F1. Six models provide 15 comparisons, and the p-values are adjusted using the Holm procedure (Bonferroni threshold 0.05/15 = 0.0033 for the smallest p-value). Effect sizes are presented as the difference in MCC with interval and Cliff's delta over seeds. The decision rule for each question is given in Table V.

TABLE V. PRE-SPECIFIED ANALYSIS PLAN. CI: 95% CONFIDENCE INTERVAL; AURC: AREA UNDER THE RISK-COVERAGE CURVE; K = 15 PAIRWISE COMPARISONS FOR SIX MODELS

RQ	Quantity	Estimator and test	Decision rule
RQ4	Detection performance, six models and three baselines	MCC, macro-F1 with CI over 5 seeds; paired bootstrap and exact McNemar; Holm correction (k = 15)	A model is better than another only if the adjusted p < 0.05 and the MCC difference interval excludes 0
RQ4	Source shift	Delta MCC between within-source and leave-one-source-out, per model, with CI	Report each delta; no directional claim without an interval excluding 0
RQ3	Calibration	ECE (15 bins), Brier score, NLL with bootstrap CI, before and after scaling	Scaling is retained only if ECE decreases with an interval excluding 0
RQ3	Selective prediction	AURC and error at 80% coverage; paired bootstrap against the best single model	Agreement is useful only if it lowers error at 80% coverage with an interval excluding 0
RQ3	Conformal coverage	Empirical coverage with Wilson CI, within source and held-out source, alpha in {0.05, 0.10, 0.20}	Within-source coverage interval must contain 1 − alpha; shifted-source coverage is reported without guarantee
RQ4	Evasion robustness	Attack success rate per operator and family, Wilson CI	Report all rates; a requirement SR4 or SR11 fails if flag recall is below the pre-registered threshold
RQ4	Behavioural validity	Failure rate per capability and family, Wilson CI	Report all rates; capabilities with failure above 0.20 require a documented mitigation
RQ1	Security requirements	Pass or fail per test	Every requirement must pass or carry a documented exception

D. Behavioral and evasion test suite

Stage E6 tests the behavioral validity and evasion robustness using controlled test cases [38], [39]. The capabilities and the generating operators are given in Table VI. The evasion operators apply to published attack classes [6] – [8], and the obfuscation cases involve spacing out letters and substituting symbols, which are in the source data. The half-width of the Wilson interval for a failure rate near 0.5 is about 0.096 for 100 cases and about 0.168, too large for a failure rate claim, for 30 cases, so 100 cases are the minimum. Cohen's kappa is reported, and each case is validated by two annotators.

TABLE VI. TEST SUITE SPECIFICATION FOR BEHAVIOURAL AND EVASION. EXAMPLES ARE PROVIDED AND NO ABUSIVE TEXT IS REPRODUCED

Capability	Generating operator	Example form	Cases per family
Negation	Insert a negator before an abusive predicate	You are not [INSULT]	100
Reported abuse	Quote or report abuse without endorsing it	He called me [INSULT] today	100
Sarcasm cue	Positive wording with an abusive cue	Great job, [INSULT]	100
Character obfuscation	Letter spacing, symbol and digit substitution	[I N S U L T], [1NSULT]	100
Word-boundary removal	Remove spaces between words	youare[INSULT]	100
Dilution	Append innocuous words	[INSULT] love love love	100
Typographical noise	Random character edits at a fixed rate	[INSUUT], [INSLT]	100
Identity terms	Minimal pairs differing only in identity term	I am [IDENTITY]	100 pairs
Benign profanity	Non-targeted profanity	This traffic is [PROFANITY]	100

E. Audit of the prototype's reported results

The performance table, the confusion-matrix cell counts, and the ROC curve legend are the three artifacts of the prototype report used in the audit. The values of the cell counts were read from the figure, and the summary annotations in the figure (accuracy, precision, recall, F1) were recomputed from these cells to check the value. Wilson intervals are displayed for accuracy, true-positive rate, and false-positive rate. The Matthews correlation coefficient is associated with a percentile interval obtained by 10,000 parametric bootstrap samples of the four cells. An exact binomial test is used to compare with the always-negative accuracy on the same test set. AUC values are tested against 0.5 with the Hanley and McNeil standard error under the null [49], assuming that the ROC test set is one of the confusion matrices. Prevalence is calculated from accuracy, precision, and recall and implied by the performance table. The audit script is included in the supplementary material.

F. Synthetic validation of the abstention mechanism

The synthetic study examines the abstention mechanism with a known data generating process. Twenty features are selected for each replication, six of which are informative; the prevalence of bullying is 0.35, and the class-conditional shifts for the informative features are ±0.42. The shifted source attenuates the informative features by 0.8, introduces noise with standard deviation 0.7, and also reverses some of the cues on three of the informative features, and adds features that are only present in the shifted source. Six learners stand in for the model families: logistic regression, linear SVM, Gaussian Naive Bayes, a random forest with 80 trees, a multilayer perceptron with 32 hidden units and histogram gradient boosting. Learners are trained on 3,000 items from the first source, and calibrated on 1,500 additional items using Platt scaling. They are tested on 2,000 in-source items and 2,000 shifted items. The study adopts 100 replications with seeds 1000 to 1099, 95% t-intervals over replications, and the Wilcoxon signed rank test for paired differences in error at 80% coverage. These data are not actual. The study does not make any claim about the performance of any model on abusive-language text, but it validates the behavior of the mechanism.

Answer to RQ4. The data audit, grouped, source-held-out and temporal splits, equal-budget training of 6 families with baselines, calibration, agreement and abstention analysis, behavioural and evasion suite (9 capabilities, at least 100 cases per capability), security tests and pre-specified statistical plan with Holm correction on 15 comparisons comprise 10 protocol stages. Each of the research questions can be answered or refuted with the executed protocol as stated in Table V, which contains a decision rule for each quantity.

VI. Results

A. Audit of the prototype's reported results (RQ2)

Table VII provides the metrics re-computed from the counts in the confusion matrix provided in the prototype report [17]. Each of the six matrices sums to 1,000 texts, of which 387 (38.7%, Wilson 95% CI 35.7% to 41.8%) belong to the bullying class. The first four recomputed metrics repeat the in-figure annotations to three decimals for each model. The recomputed accuracy is between 0.384 (BERT) and 0.500 (CNN). On the same test set, the always-negative classifier achieves an accuracy of 0.613. BERT and RoBERTa label 99.1% and 98.7% of the texts as bullying, with specificities of 0.005 and 0.010. For all models, the MCC ranges from −0.055 to 0.000 with all 95% intervals containing 0.

TABLE VII. METRICS RECOMPUTED FROM THE CONFUSION MATRICES (N = 1,000; 387 BULLYING TEXTS). ACC: ACCURACY; TPR: TRUE-POSITIVE RATE; FPR: FALSE-POSITIVE RATE; MCC: MATTHEWS CORRELATION COEFFICIENT. 95% CONFIDENCE INTERVALS (CI) (WILSON FOR PROPORTIONS AND PERCENTILE BOOTSTRAP FOR MCC) ARE PROVIDED BY THE BRACKETS. POS. RATE: PERCENTAGE OF TEXTS BEING CLASSIFIED AS BULLYING

Model	Acc [95% CI]	TPR [95% CI]	FPR	MCC [95% CI]	Pos. rate
SVM	0.417 [0.387, 0.448]	0.742 [0.696, 0.783]	0.788	-0.054 [-0.116, 0.009]	0.770
Naive Bayes	0.408 [0.378, 0.439]	0.876 [0.839, 0.905]	0.887	-0.017 [-0.081, 0.044]	0.883
CNN	0.500 [0.469, 0.531]	0.499 [0.449, 0.548]	0.499	0.000 [-0.061, 0.061]	0.499
LSTM	0.479 [0.448, 0.510]	0.535 [0.485, 0.584]	0.556	-0.021 [-0.082, 0.042]	0.548
BERT	0.384 [0.354, 0.415]	0.984 [0.967, 0.993]	0.995	-0.055 [-0.113, 0.011]	0.991
RoBERTa	0.386 [0.356, 0.417]	0.982 [0.963, 0.991]	0.990	-0.036 [-0.097, 0.030]	0.987

Table VIII examines the performance table of the same report. The results for F1 recomputed from precision and recall agree to within 0.0001 with the reported F1 for all rows, and the prevalence implied from accuracy, precision, and recall is stable for the 6 models, ranging between 0.331 and 0.336. This value is not within the 95% interval of the prevalence in the confusion matrix (0.357, 0.418). The performance table is thus for a test set whose positive share differs from the one in the confusion matrices. On its own terms, the table shows the models to be 1.8 to 15.5 percentage points better than the always-negative accuracy shown in the table by the models themselves.

TABLE VIII. CONSISTENCY CHECKS OF THE PERFORMANCE TABLE. THE ACCURACY, PRECISION, AND RECALL ARE REPORTED, AND THE IMPLIED PREVALENCE IS CALCULATED. MARGIN: DIFFERENCE BETWEEN THE REPORTED ACCURACY AND THE ALWAYS NEGATIVE ACCURACY (ONE MINUS THE IMPLIED PREVALENCE) IN PERCENTAGE POINTS.

Model	Reported F1	Recomputed F1	Implied prevalence	Always-negative acc.	Margin (pp)
SVM	0.627	0.627	0.331	0.669	5.4
Naive Bayes	0.670	0.670	0.336	0.664	9.0
CNN	0.617	0.617	0.331	0.669	1.8
LSTM	0.642	0.642	0.331	0.669	4.2
BERT	0.727	0.727	0.331	0.669	15.5
RoBERTa	0.686	0.686	0.331	0.669	13.7

Table IX and Fig. 10 compare the recomputed accuracy with the always-negative accuracy (0.613) and the AUC values in the ROC figure (0.5). The exact binomial test shows that, for all six models, equality of accuracy with 0.613 is rejected at all $p < 0.001$ (downward). The 95% intervals of the AUC values contain 0.5 for all models, and none of the six tests is significantly different from chance (p-values range from 0.110 to 0.957). The six accuracy tests and six AUC tests were not multiplicity adjusted. The Bonferroni-corrected p-values are all below 0.0042 (0.05/12), and there is no p-value below 0.05 for the AUC; therefore, the conclusions regarding accuracy and AUC are unchanged.

TABLE IX. COMPARES AGAINST STANDARD VALUES. BINOMIAL P: EXACT TWO-SIDED TEST OF THE RECOMPUTED ACCURACY AGAINST THE ALWAYS NEGATIVE ACCURACY OF 0.613. AUC: AREA UNDER THE ROC CURVE (PRINTED IN THE REPORT LEGEND) CALCULATED WITH THE HANLEY AND MCNEIL 95% INTERVAL AND Z-TEST AGAINST 0.5.

Model	Accuracy	Binomial p	AUC [95% CI]	z vs 0.5	p
SVM	0.417	< 0.001	0.470 [0.433, 0.507]	-1.60	0.110
Naive Bayes	0.408	< 0.001	0.501 [0.464, 0.538]	0.05	0.957
CNN	0.500	< 0.001	0.498 [0.461, 0.535]	-0.11	0.915
LSTM	0.479	< 0.001	0.486 [0.449, 0.523]	-0.75	0.455
BERT	0.384	< 0.001	0.521 [0.484, 0.558]	1.12	0.263
RoBERTa	0.386	< 0.001	0.525 [0.488, 0.562]	1.33	0.182

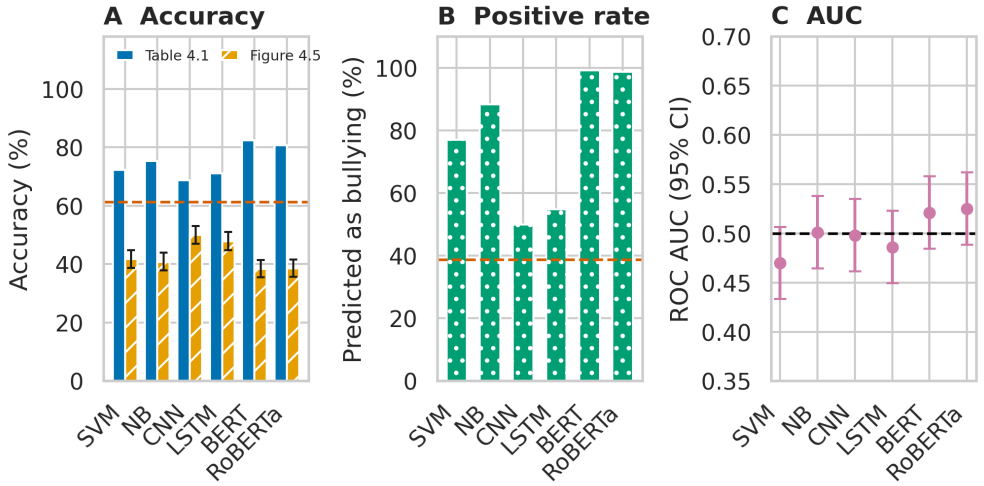

Fig. 10. Recomputation of reported results for prototype. Panel A: accuracy for the performance table (blue) and accuracy recomputed from the confusion matrices with 95% Wilson intervals (orange, hatched); the dashed line represents the always negative accuracy of 0.613. Panel B: percentage of 1,000 test texts classified as bullying with the dashed line indicating the prevalence of 0.387. Panel C: AUC with 95% intervals (dashed line represents chance 0.5).

Each model is plotted in ROC space in Fig. 11. All the 6 points fall inside their 95% intervals of the chance diagonal. The transformer models are located in the upper-right corner (true-positive rate 0.984 and 0.982; false-positive rate 0.995 and 0.990), indicating that they label almost all texts as bullying. The classical models lie at FPR 0.788 (SVM) and 0.887 (Naive Bayes) while the CNN and LSTM are positioned close to the diagonal.

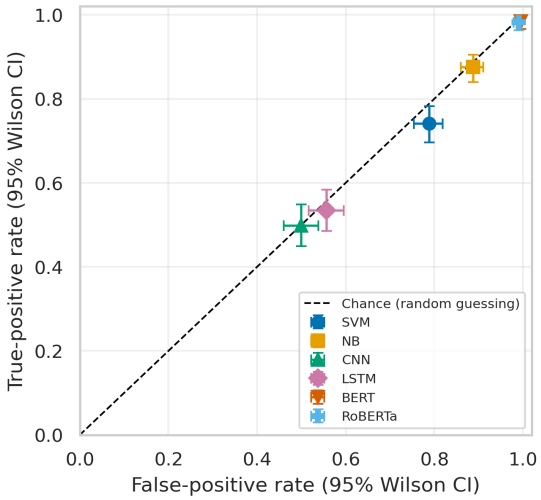

Fig. 11. Operating points of the six models in ROC space, with 95% Wilson intervals on both axes, based on the counts in the confusion matrices. Chance performance is indicated by the dashed diagonal.

Thus, the error profile is skewed towards false positives for the four models that have false-positive rates > 0.55, and towards an roughly equal mix of errors for the CNN. There are two more observations from the prototype interface. For an abusive example (Fig. 7c) and a benign example, the SVM panel displays exactly 50.00% and the confidence displayed for a benign input is the probability of the bullying class. From the two observations with a single input, the audit cannot conclude if the deployed models are the ones that created the confusion matrices.

Answer to RQ2. The result artifacts in the prototype are inconsistent with each other. The confusion matrices give accuracies of 0.384 to 0.500, all below the always-negative accuracy of 0.613 (binomial $p < 0.001$), and the AUC values of 0.470 to 0.525 do not differ from 0.5 (p from 0.110 to 0.957), whereas the performance table reports 68.69% to 82.44% and implies a different test-set prevalence (0.331 to 0.336 against 0.387). So the evidence reported does not support the detection claims and the report does not allow a determination of which artifact is correct.

B. Synthetic validation of calibrated agreement and conformal abstention (RQ3)

The synthetic study of over 100 replications is presented in Table X and Fig. 12. In the first source, the accuracy of the ensemble using mean calibrated probability is 0.856 [0.854, 0.857], and the best single learner is 0.857 [0.855, 0.858]. Both dropped to approximately 0.633 with source shift. At 80% coverage the error among accepted items was 8.46 [8.32, 8.59]% for the ensemble, 8.34 [8.20, 8.48]% for the best single learner, and 8.63 [8.49, 8.78]% for the vote share within the source. The difference between the ensemble and the best single learner was 0.12 [0.05, 0.19] percentage points (Wilcoxon $p < 0.001$; the positive value means that the ensemble had a higher error). Under shift the corresponding difference was -0.11 [-0.23, 0.01] points ($p = 0.249$). The mean calibrated probability produced a lower error than the vote share in both conditions (differences of -0.18 [-0.21, -0.14] and -0.19 [-0.26, -0.11] points; both $p < 0.001$).

The empirical coverage for the split-conformal sets at $\alpha = 0.10$ was 0.901 [0.898, 0.903] within the source, which includes the nominal 0.90. Under source shift, the coverage fell to 0.712 [0.708, 0.716]. The percentage of items sent to the policy for review increased from 10.4% to 16.4% of items with a non-singleton set. For the items with a singleton set, the error increased from 11.1% to 34.4%.

TABLE X. SYNTHETIC VALIDATION RESULTS (100 REPLICATIONS, TEST SETS OF 2000 ITEMS/CONDITION). VALUES ARE MEANS WITH 95% T-INTERVALS OVER REPLICATIONS. ERR@80: ERROR IN ITEMS ACCEPTED AT 80% COVERAGE. DIFFERENCES ARE PERCENTAGE POINTS (POSITIVE IS WORSE THAN THE COMPARATOR). SYNTHETIC, NOT 6 MODEL FAMILIES, STAND-INS FOR THE LEARNERS.

Quantity	In-source	Shifted source
Accuracy, ensemble (mean calibrated probability)	0.856 [0.854, 0.857]	0.633 [0.630, 0.635]
Accuracy, best single learner	0.857 [0.855, 0.858]	0.632 [0.630, 0.635]
Err@80 (%), ensemble	8.46 [8.32, 8.59]	33.94 [33.64, 34.24]
Err@80 (%), best single learner	8.34 [8.20, 8.48]	34.04 [33.76, 34.33]
Err@80 (%), vote share	8.63 [8.49, 8.78]	34.12 [33.83, 34.42]
Ensemble minus best single (pp); Wilcoxon p	0.12 [0.05, 0.19]; < 0.001	-0.11 [-0.23, 0.01]; 0.249
Vote share minus best single (pp); Wilcoxon p	0.29 [0.22, 0.37]; < 0.001	0.08 [-0.06, 0.22]; 0.120
Conformal coverage at $\alpha = 0.10$ (target 0.90)	0.901 [0.898, 0.903]	0.712 [0.708, 0.716]
Share of non-singleton sets (abstention)	0.104 [0.099, 0.109]	0.164 [0.157, 0.171]

Error among singleton sets	0.111 [0.108, 0.113]	0.344 [0.341, 0.347]

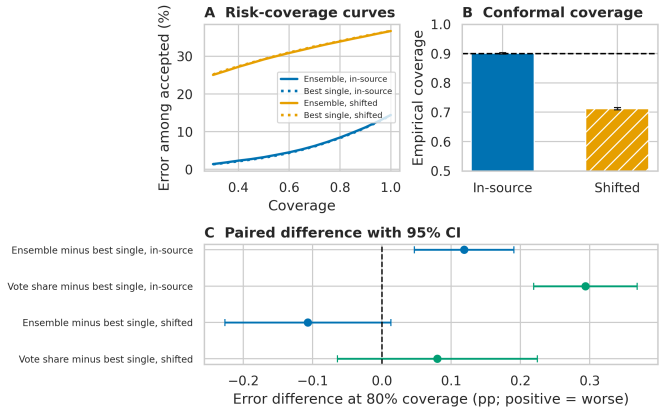

Fig. 12. Abstention mechanism synthetically validated. Panel A: mean risk-coverage curves for the ensemble (solid) and the best single learner (dotted) in the first source (blue) and the shifted source (orange). Panel B: empirical conformal coverage at $\alpha = 0.10$, 95% intervals; the dashed line is the nominal 0.90. Panel C: Difference of the error at 80% coverage relative to the best single learner, with 95% intervals; positive differences mean that the error is larger than the best single learner.

Answer to RQ3. In the synthetic study, the split-conformal sets maintained their nominal level of coverage in the source (0.901, 95% CI [0.898, 0.903]) but not in the shift (0.712, 95% CI [0.708, 0.716]). Consequently, coverage under shift needs to be monitored, and cannot be assumed. Averaging calibrated probabilities reduced error compared with vote share ($p < 0.001$ in both conditions) but did not improve on the best single learner within the source (difference +0.12 percentage points, $p < 0.001$) or under shift (−0.11, $p = 0.249$), so agreement adds value only if learner errors differ.

VII. Discussion

A. What the audit establishes and what it does not

The audit reveals that the evidence reported does not support the claims for detection performance and that there are two consistent explanations for the difference. In the first, the performance table and the confusion matrices are derived from different runs of the evaluation or test sets, which is supported by the difference in implied prevalence (0.331 to 0.336 and 0.387). In the second, faulty prediction outputs, e.g., label-alignment or class-mapping fault, were used to generate the confusion matrices and ROC curves. The audit does not distinguish between the two explanations, nor between the fact that the models are inherently at random. The performance table is self-consistent and, on its own, would result in all six models being above the negative accuracy. The practical impact is also a procedural one: step E9 of the protocol and requirement SR12 require the production of results, confusion matrices, and curves from one prediction file that is saved. This result, that one of six models is only 1.8 percentage points above the trivial baseline, also demonstrates the need for trivial baselines in every results table [14].

B. Security interpretation of calibrated abstention

Abstention is a response to two of the threats in Table II. In evasion (T1), the inputs are given close to the decision boundary post perturbation and tend to result in disagreement and non-singleton sets which are not silently accepted but sent to review. The same mechanism restricts automated action on ambiguous text for weaponized false positives (T8). The synthetic study also demonstrates the limitation of the mechanism. When the input distribution changes, e.g., due to an evasion campaign or a new platform, the coverage drops from 0.90 to 0.71 and the error among predictions that were accepted increases from 11% to 34%, while the abstention rate only increased by 6 percentage points. Thus, a deployed service should have a monitoring control that provides a best guess estimate of coverage based on a regularly refreshed labeled audit sample and triggers an alert if $1 - \alpha$ is not contained in the Wilson interval of the empirical coverage. Abstentions are a good but not a comprehensive drift measure, as they also increased slightly while coverage fell. This discovery is in line with the observation that it is exactly exchangeable, which the coverage guarantee requires [36] and which is violated by an adaptive adversary [6], [7].

C. Agreement versus calibrated confidence

Averaging calibrated probabilities dominated vote share in both conditions, in line with the information lost when probabilities are reduced to votes. The ensemble did not outperform the best single learner, which is understandable as the learners were trained on the same features and were prone to making correlated errors. Whether the errors of the heterogeneous model families, such as transformers fine-tuned on raw text, are sufficiently diverse on real abusive-language data is an empirical issue. As such, the protocol considers error overlap and pairwise agreement prior to any statement about the multi-model design, and the hypothesis for *RQ3* is stated conditionally with respect to error diversity. The finding is a warning to the prototype; its dashboard shows consensus among six models as reassurance, and it is not known if consensus tracks correctness.

D. Relation to prior work

Related work provides a practical example of the validity concerns raised for the field [13], [14], [30], and the design implements their suggestions as requirements and protocol stages. The brittleness leads to the normalizer, the evasion operators, and the need to retest following each model upgrade [6] – [8]. The mechanism is provided by conformal prediction for toxic-content classification [37], and this contribution is its incorporation into a threat-modelled service, with an explicit source-shift behavior analysis. The design does not have any robustness claims. It details the method of measuring robustness.

E. Implications for operators

Operators should ask that all reported metrics include the trivial baseline value, the size of the test set, a confidence interval, and a reference to the prediction file. They should take the abstention share as a quantity for capacity planning, as in the synthetic shifted condition, 16.4% of the items needed to be reviewed. They should also consider scores per model as sensitive outputs and only expose calibrated decisions to untrusted callers, as in SR5.

VIII. Threats to validity

The threats are presented based on the 4-type taxonomy [50] and an additional type related to security validity as listed in Table XI.

Table XI. Threats to validity, mitigations and residual risks.

Type	Threat	Mitigation	Residual risk
Internal	Confusion-matrix counts were read from a printed figure	Recomputed accuracy, precision, recall and F1 match the in-figure annotations to three decimals for all six models	A reading error in a cell that the annotations do not constrain is unlikely
Internal	The AUC test assumes the ROC curves use the 1,000-text test set of the confusion matrices	Assumption stated; intervals of about ±0.037 would widen if N were smaller	If N differs, intervals change but the conclusion that AUC is near 0.5 depends on the reported values
Internal	Synthetic generator parameters were fixed in advance and not varied	Seeds and parameters are reported; the script is supplied	Sensitivity to the generator is unexplored
External	One prototype report was audited; the synthetic data are Gaussian with learners that are not the six model families	Claims are limited to the audited artefacts and to mechanism behaviour	The error structure of real text may differ; protocol stages E4 and E5 address this
External	The target corpus is English and multi-source	Source-held-out evaluation is part of the protocol	Other languages and platforms are not covered
Construct	Message-level labels do not capture repetition, intent or power imbalance	The construct is stated as message-level abuse detection	Cyberbullying in the full sense is not measured
Construct	Agreement and conformal sets are proxies for correctness and uncertainty	Marginal coverage is reported with intervals; error overlap is measured	Per-subgroup coverage is not guaranteed
Conclusion	Twelve audit tests were not adjusted for multiplicity	All conclusions are unchanged under a Bonferroni threshold of 0.0042	None identified
Conclusion	Wilcoxon tests over 100 simulated replications can detect very small differences	Differences are reported in percentage points with intervals; practical relevance is judged on the magnitude	A statistically significant difference of 0.1 to 0.3 points has no operational relevance
Security	The threat model was derived by analysis and not by penetration testing; adversaries are black-box	Requirements are paired with executable tests (E6, E7)	White-box and adaptive attacks are not covered

IX. Ethics, privacy, and responsible use

A published prototype report is analyzed, and synthetic data is generated. No human data are collected, and no human subjects are involved. The target dataset is a collection of public posts and comments from several platforms [21], and the use of this dataset is subject to the licenses and terms of the target platforms.

The service deals with personal text. This means that no raw text is kept by default, pseudonymous audit events are stored with a retention limit, and exports are redacted (SR8). Operators continue to be liable for compliance with the data-protection law applicable to them, such as the General Data Protection Regulation [42] and the Nigeria Data Protection Act [43], and with obligations regarding platforms under the Digital Services Act [2]. Verdicts are for use in human review only and should not be used for sanctions. The test suite includes identity-term and benign-profanity cases, as abusive-language classifiers can be adversely affected by dialect and identity terms [40]. Documentation of evasion operators might help adversaries, but it is not a new attack, and operators are already available [6], [7].

X. Conclusion and future work

This paper presented a multi-model design of a cyberbullying detection service based on a threat model. Eight threats are mapped to twelve verifiable requirements, and the architecture is a combination of per-model calibration, an agreement score, split-conformal prediction sets, and an abstain-to-human policy, with use-case, class, sequence, and interface designs. The results of an audit of an earlier prototype indicate that the 68.69%-82.44% accuracy may not be reconcilable with the confusion matrices of 38.4%-50% and the ROC curves indistinguishable from

chance. A synthetic study revealed that under source shift, conformal coverage drops to 0.712, and conformal coverage is only beneficial in the case of learner errors that are different.

The paper does not do a design evaluation based on real data, nor does it claim anything about the relative design quality of the six model families. The next phase performs the protocol phases E0 to E9 on the multi-source corpus with leave-one-source-out splits, equal-budget training, calibration, agreement/abstention analysis, the behavioral and evasion suite, and the security tests. Future research will include a guard-model baseline and adversarial training [8] as well as a coverage-monitoring component with a newly collected audit sample and a reviewer-workload study of the abstention policy. An extension to repetition and intent with annotated resources [22] would be a step towards cyberbullying detection in the full sense rather than just message-level abuse detection.

REFERENCES

[1] B. Yang *et al.*, "The consequences of cyberbullying and traditional bullying victimization among adolescents: Gender differences in psychological symptoms, self-harm and suicidality," *Psychiatry Res.*, vol. 306, Art. no. 114219, 2021.

[2] European Parliament and Council of the European Union, "Regulation (EU) 2022/2065 on a single market for digital services (Digital Services Act)," *Off. J. Eur. Union*, vol. L 277, Oct. 2022.

[3] National Institute of Standards and Technology, "Artificial Intelligence Risk Management Framework (AI RMF 1.0)," NIST, Gaithersburg, MD, USA, Rep. NIST AI 100-1, 2023.

[4] S. D. Okegbile and I. P. Gambo, "Artificial intelligence-driven security framework for internet of things-enhanced digital twin networks," *Internet Things*, Art. no. 101564, 2025.

[5] R. Massenon, I. Gambo, and J. A. Khan, "Unveiling hidden permissions: An LLM framework for detecting privacy and security concerns in AI mobile apps reviews," *Autom. Softw. Eng.*, vol. 33, Art. no. 42, 2026, doi: 10.1007/s10515-025-00567-9.

[6] H. Hosseini, S. Kannan, B. Zhang, and R. Poovendran, "Deceiving Google's Perspective API built for detecting toxic comments," arXiv:1702.08138, 2017.

[7] T. Gröndahl, L. Pajola, M. Juuti, M. Conti, and N. Asokan, "All you need is 'love': Evading hate speech detection," in *Proc. 11th ACM Workshop Artif. Intell. Secur.*, 2018, arXiv:1808.09115.

[8] M. Juarez *et al.*, "Analyzing adversarial strategies and countermeasures for cyberbullying detection," in *Social, Cultural, and Behavioral Modeling (SBP-BRiMS 2025)* (Lecture Notes in Computer Science). Springer, 2025, doi: 10.1007/978-3-032-07715-8_9.

[9] F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Stealing machine learning models via prediction APIs," in *Proc. 25th USENIX Secur. Symp.*, 2016, pp. 601–618.

[10] E. Wallace, T. Zhao, S. Feng, and S. Singh, "Concealed data poisoning attacks on NLP models," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Lang. Technol. (NAACL-HLT)*, 2021, pp. 139–150.

[11] M. Deng, K. Wuyts, R. Scandariato, B. Preneel, and W. Joosen, "A privacy threat analysis framework: Supporting the elicitation and fulfillment of privacy requirements," *Requirements Eng.*, vol. 16, no. 1, pp. 3–32, 2011.

[12] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *Proc. IEEE Symp. Secur. Privacy (SP)*, 2017, pp. 3–18.

[13] H. Rosa *et al.*, "Automatic cyberbullying detection: A systematic review," *Comput. Hum. Behav.*, vol. 93, pp. 333–345, 2019.

[14] C. Emmery *et al.*, "Current limitations in cyberbullying detection: On evaluation criteria, reproducibility, and data scarcity," *Lang. Resour. Eval.*, vol. 55, pp. 597–633, 2021.

[15] R. Massenon, I. Gambo, and J. A. Khan, "Toward an automated cross-multimodal verification of mobile app bug fixes integrating user feedback, developer responses, changelogs, and UI visual analysis," *Inf. Softw. Technol.*, Art. no. 107996, 2025, doi: 10.1016/j.infsof.2025.107996.

[16] R. Massenon, I. Gambo, J. A. Khan, C. Agbonkhese, and A. Alwadain, "'My AI is lying to me': User-reported LLM hallucinations in AI mobile apps reviews," *Sci. Rep.*, vol. 15, no. 1, Art. no. 30397, 2025.

[17] C. O. Balogun, "Design and implementation of a cyberbullying detection system," undergraduate project report, Dept. Comput. Sci. Eng., Obafemi Awolowo Univ., Ile-Ife, Nigeria, 2025.

[18] I. Gambo, R. Massenon, C.-C. Lin, R. O. Ogundokun, S. Agarwal, and W. Pak, "Enhancing user trust and interpretability in AI-driven feature request detection for mobile app reviews: An explainable approach," *IEEE Access*, vol. 12, pp. 114023–114045, 2024.

[19] P. K. Smith, J. Mahdavi, M. Carvalho, S. Fisher, S. Russell, and N. Tippett, "Cyberbullying: Its nature and impact in secondary school pupils," *J. Child Psychol. Psychiatry*, vol. 49, no. 4, pp. 376–385, 2008.

[20] E. Wulczyn, N. Thain, and L. Dixon, "Ex Machina: Personal attacks seen at scale," in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 1391–1399.

[21] F. Elsafoury, "Cyberbullying datasets," Mendeley Data, V1, 2020, doi: 10.17632/jf4pzyvnpj.1.

[22] N. Ejaz, F. Razi, and S. Choudhury, "Towards comprehensive cyberbullying detection: A dataset incorporating aggressive texts, repetition, peerness, and intent to harm," *Comput. Hum. Behav.*, Art. no. 108123, 2023, doi: 10.1016/j.chb.2023.108123.

[23] N. Chamidah and R. Sahawaly, "Comparison support vector machine and naive Bayes methods for classifying cyberbullying in Twitter," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, vol. 7, no. 2, p. 338, 2021.

[24] F. Elsafoury, S. Katsigiannis, Z. Pervez, and N. Ramzan, "When the timeline meets the pipeline: A survey on automated cyberbullying detection," *IEEE Access*, vol. 9, 2021.

[25] B. Ogunleye and B. Dharmaraj, "The use of a large language model for cyberbullying detection," *Analytics*, vol. 2, no. 3, pp. 694–707, 2023.

[26] T. H. Teng, K. D. Varathan, and F. Crestani, "A comprehensive review of cyberbullying-related content classification in online social media," *Expert Syst. Appl.*, vol. 244, Art. no. 122644, 2024.

[27] A. Muminovic, "Moderating harm: Benchmarking large language models for cyberbullying detection in YouTube comments," arXiv:2505.18927, 2025.

[28] A. G. Philipo, D. S. Sarwatt, J. Ding, M. Daneshmand, and H. Ning, "Assessing text classification methods for cyberbullying detection on social media platforms," arXiv:2412.19928, 2025.

[29] P. Yi and A. Zubiaga, "Cyberbullying detection across social media platforms via platform-aware adversarial encoding," in *Proc. Int. AAAI Conf. Web Soc. Media*, 2022, arXiv:2204.00334.

[30] A. Arango, J. Pérez, and B. Poblete, "Hate speech detection is not as easy as you may think: A closer look at model validation," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2019, pp. 45–54.

[31] C. Emmery, Á. Kádár, G. Chrupała, and W. Daelemans, "Cyberbullying classifiers are sensitive to model-agnostic perturbations," arXiv:2201.06384, 2022.

[32] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, vol. 70, pp. 1321–1330.

[33] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*, A. J. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans, Eds. Cambridge, MA, USA: MIT Press, 1999, pp. 61–74.

[34] M. P. Naeini, G. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using Bayesian binning," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015.

[35] Y. Geifman and R. El-Yaniv, "Selective classification for deep neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.

[36] A. N. Angelopoulos and S. Bates, "Conformal prediction: A gentle introduction," *Found. Trends Mach. Learn.*, vol. 16, no. 4, pp. 494–591, 2023.

[37] G. Villate-Castillo, J. Del Ser, and B. Sanz, "Conformal prediction for toxic content classification," arXiv:2411.04090, 2024.

[38] M. T. Ribeiro, T. Wu, C. Guestrin, and S. Singh, "Beyond accuracy: Behavioral testing of NLP models with CheckList," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 4902–4912.

[39] P. Röttger, B. Vidgen, D. Nguyen, Z. Waseem, H. Margetts, and J. Pierrehumbert, "HateCheck: Functional tests for hate speech detection models," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics 11th Int. Joint Conf. Natural Lang. Process.*, 2021, pp. 41–58.

[40] M. Sap, D. Card, S. Gabriel, Y. Choi, and N. A. Smith, "The risk of racial bias in hate speech detection," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 1668–1678.

[41] M. Mitchell *et al.*, "Model cards for model reporting," in *Proc. Conf. Fairness, Accountability, Transparency*, 2019, pp. 220–229.

[42] European Parliament and Council of the European Union, "Regulation (EU) 2016/679 on the protection of natural persons with regard to the processing of personal data (General Data Protection Regulation)," *Off. J. Eur. Union*, vol. L 119, May 2016.

[43] Federal Republic of Nigeria, "Nigeria Data Protection Act, 2023," 2023.

[44] F. Khan and I. P. Gambo, "Incorporating privacy requirements in smart communities for older adults: A research vision," in *Proc. 17th Int. Conf. Softw. Technol.*, Lisbon, Portugal, Jul. 2022, pp. 242–249.

[45] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Lang. Technol.*, 2019, pp. 4171–4186.

[46] Y. Liu *et al.*, "RoBERTa: A robustly optimized BERT pretraining approach," arXiv:1907.11692, 2019.

[47] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, Art. no. 6, 2020.

[48] L. D. Brown, T. T. Cai, and A. DasGupta, "Interval estimation for a binomial proportion," *Statist. Sci.*, vol. 16, no. 2, pp. 101–133, 2001.

[49] J. A. Hanley and B. J. McNeil, "The meaning and use of the area under a receiver operating characteristic (ROC) curve," *Radiology*, vol. 143, no. 1, pp. 29–36, 1982.

[50] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in Software Engineering*. Springer, 2012.