TCLens: Towards Toxicity Tags Aggregation of Massive Labels Generated by Content Moderation for AIGC

Bingtao Chang
btchang@iflytek.com
iFLYTEK Security Laboratory
Hefei, China

Weiping Wen
weipingwen@pku.edu.cn
Peking University
Beijing, China

Xiaojie Wu
xjwu@iflytek.com
iFLYTEK Security Laboratory
Hefei, China

Siyang Cheng
sycheng9@iflytek.com
iFLYTEK Security Laboratory
Hefei, China

Jianchun Jiang
jianchun@iscas.ac.cn
Institute of Software,
Chinese Academy of Sciences
Beijing, China

Rui Mei*
ruimei@pku.edu.cn
iFLYTEK Security Laboratory
Hefei, China
Software Security Research Group,
Peking University
Beijing, China

ABSTRACT

The recent boost of artificial intelligence represented by Large Language Models (LLMs) is surging. Due to the outstanding performance of LLMs, AI-Generated Content (AIGC) has also made important progress in multimodal knowledge creation referring to text, image, audio, and video. However, the security, privacy, and ethical risks associated with AIGC (e.g., fake news, social engineering attacks, and toxic content) have deeply weakened the compliance of AIGC. Although existing content moderation solutions can filter out several types of toxic content, the audit performance of different vendors and techniques are of varying quality. Some AIGC service providers improve the moderation effectiveness by introducing multiple sources of audit vendors. Due to the lack of general content moderation standards and taxonomy, the labels of multi-source moderation vendors vary greatly. To this end, We propose a novel massive label aggregation approach for content moderation named TCLens. First, we collect results of multi-vendor content moderation engines for building massive toxic labels for AIGC. Then, we introduce an ontology for better tagging with the capability of automatic updating and vendor-agnostic. Finally, we implement a prototype of TCLens. Our evaluation demonstrates that it outperforms single-source tagging and existing SOTA solutions.

CCS CONCEPTS

• **Security and privacy** → **Social aspects of security and privacy**; • **Information systems** → **Content analysis and feature selection**.

*Corresponding author.

CSAI 2024, December 06–08, 2024, Beijing, China

ACM ISBN XXX-X-XXXX-XXXX-X/XX/XX…$XXX.00
https://doi.org/XXXXXXX.XXXXXXX

KEYWORDS

information content moderation, toxicity tags, labels aggregation, AIGC

ACM Reference Format:
Bingtao Chang, Weiping Wen, Xiaojie Wu, Siyang Cheng, Jianchun Jiang, and Rui Mei. 2024. TCLens: Towards Toxicity Tags Aggregation of Massive Labels Generated by Content Moderation for AIGC. In *Proceedings of 2024 8th International Conference on Computer Science and Artificial Intelligence (CSAI 2024).* ACM, New York, NY, USA, 8 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Artificial Intelligence (AI) and AIGC have progressed rapidly in recent years, transforming various industries. AI models have achieved breakthroughs in natural language processing (NLP), computer vision, and decision-making [43]. OpenAI's language models [30], for instance, demonstrate unprecedented levels of comprehension and text generation, fueling developments in virtual assistants, content creation, and customer service. Meanwhile, AIGC has created new avenues for media and entertainment, enabling the automated generation of text, audio, video, and even complex images. Generative AI applications, like DALL-E [31] and Midjourney [27], allow creators to produce high-quality visual content in seconds, a capability once limited to skilled professionals. The global AIGC market, valued at approximately $10 billion in 2022, is expected to surpass $100 billion by 2030 [37].

Despite many benefits from AIGC, it also raises several security, privacy, compliance, and ethical concerns. One primary issue is data security, as many generative AI models are trained on vast datasets, often containing personal or sensitive information. This can lead to inadvertent data leaks or model inversion attacks, where an adversary reconstructs private training data from model outputs. Research highlights that privacy risks are particularly significant for large language models, where prompts can yield sensitive information from training data [7, 18]. Moreover, toxic content generation is a persistent issue, as AIGC systems can produce offensive, biased, or harmful language. AI models may reflect biases in the training data, resulting in outputs that perpetuate harmful stereotypes or misinformation. Despite extensive fine-tuning efforts, researchers

report that even advanced models can generate toxic content in specific contexts. Ensuring compliance with ethical guidelines in content generation is a growing technical challenge, as many tools lack sufficient filtering and moderation mechanisms [4, 20, 28].

Toxic content moderation for AIGC is a complex challenge, with industry approaches typically involving a combination of professional moderation vendors and AI classification models. Multi-source content moderation vendors provide auditing results with several labels for AIGC, while AI models are often trained on diverse, large datasets to identify offensive, harmful, or inappropriate material effectively. According to previous research, combining human moderators with automated systems reduces error rates, with hybrid approaches achieving accuracy rates above 95% in flagging toxic content [5]. The major challenge remains in the trade-off between precision and recall. Overly restrictive models may result in high false-positive rates, flagging benign content as toxic, while looser filters risk missing genuinely harmful content [2, 19, 32, 46].

To cope with the challenge of precision and recall rates of content moderation, aggregating labels from multiple sources is a common industry practice, leveraging the strengths of various auditing mechanisms to ensure AIGC compliance. However, the absence of a unified taxonomy standard and coordination across different jurisdictions presents a significant challenge in aggregating multi-source auditing labels into high-confidence content tags, which substantially limits the effectiveness of multi-source moderation mechanisms.

In this paper, we propose TCLens, a toxicity tags aggregation-based approach to content moderation for AIGC, which addresses the alignment challenge of multi-source auditing mechanisms, making the following contributions:

- We present a novel ontology of AIGC moderation tags, including taxonomy, tagging ruleset, and expansion ruleset, which provides a solid foundation for the aggregation of various content labels.
- We design a content moderation framework with two phases — inference and update — drawing inspiration from previous work in malware analysis.
- We implement a prototype of TCLens and perform a thorough evaluation. The experimental results are promising, showing that our method outperforms SOTA solutions.

2 RELATED WORK

The rapid proliferation of AIGC and LLM application has led to extensive research aimed at enhancing the precision and effectiveness of automated content moderation mechanisms to detect and label harmful content. This paper proposes a method to derive high-confidence content labels by aggregating labels from multiple content moderation sources.

Content Moderation and Label Aggregation. As AIGC becomes more prevalent, many current moderation systems operate with isolated moderation mechanisms, often combined with probabilistic machine learning models for label prediction. However, disparities in labeling standards across content moderation sources frequently result in inconsistent moderation outcomes. Past studies have addressed this challenge by introducing multi-source aggregation techniques that factor in label provenance and confidence levels [35]. In contrast to these methods, we present an inference-driven aggregation framework that not only reconciles labeling inconsistencies but also dynamically updates labels through an automated update.

Hierarchical Ontology-Based Labeling. Ontology design is also prevalent in content labeling domains like hate speech detection and misinformation tracking. Researchers have developed hierarchical labeling systems that create a taxonomy of granular levels, allowing for more granular content classification [16]. Within hierarchical labeling, labels are typically inherited between parent and child nodes, facilitating broader category coverage. Our approach integrates this principle by organizing toxicity labels into a hierarchical taxonomy, establishing intra- and inter-category rules to enable the expansion of inferred labels and generalization across specific content types. This structure adapts ontology-based methods to the flexible requirements of AIGC moderation.

Malware Labeling. Our research is inspired by advancements in malware label aggregation for malware analysis, which organizes malware families through label aggregation [15, 33, 34]. Specifically, the seminal AVclass2 [34] framework refines the initial model by introducing labeling rules and expansion techniques, leading to improved precision and recall. By constructing a taxonomy to coordinate labels from different antivirus vendors, AVclass2 mitigates issues with inconsistent naming conventions. Similarly, our approach adapts these principles for AIGC moderation by using multi-source content audit results to aggregate toxicity labels through label inference and ontology updates. This adjustment accounts for the complexity of toxic text and multimedia labels, substituting strict matching criteria used in malware labeling with semantic similarity for text-based content.

3 METHODOLOGY

This section details the design and implementation of TCLens. First, we describe the architecture and workflow of TCLens, which consists of two phases i.e. inference phase and update phase. Then, we present an opening ontology of content moderation that could be further improved by the security community. Last, we discuss the key procedure and algorithms of each phase.

3.1 Overview

We designed TCLens as a framework with two phases, namely inference phase and update phase. In the inference phase, we first collect the content auditing results from multi-source heterogeneous mechanisms, including external vendors, on-premise machine learning-based classification models, and crowdsourcing manual audit results. Due to lacking a unified ontology of AIGC taxonomy generally accepted by the industry, there is a large gap in data format and context semantics of audit labels between different sources of content moderation for the same content. The main work of this phase is to design a labeler module for aggregating multi-source auditing labels into a set of reliable tags with identifiable semantics as the foundational basis for the compliance of the whole life cycle of AIGC generation, release, and sharing.

The subsequent phase of TCLens is the update phase. It is well known that label categories for information content generated by

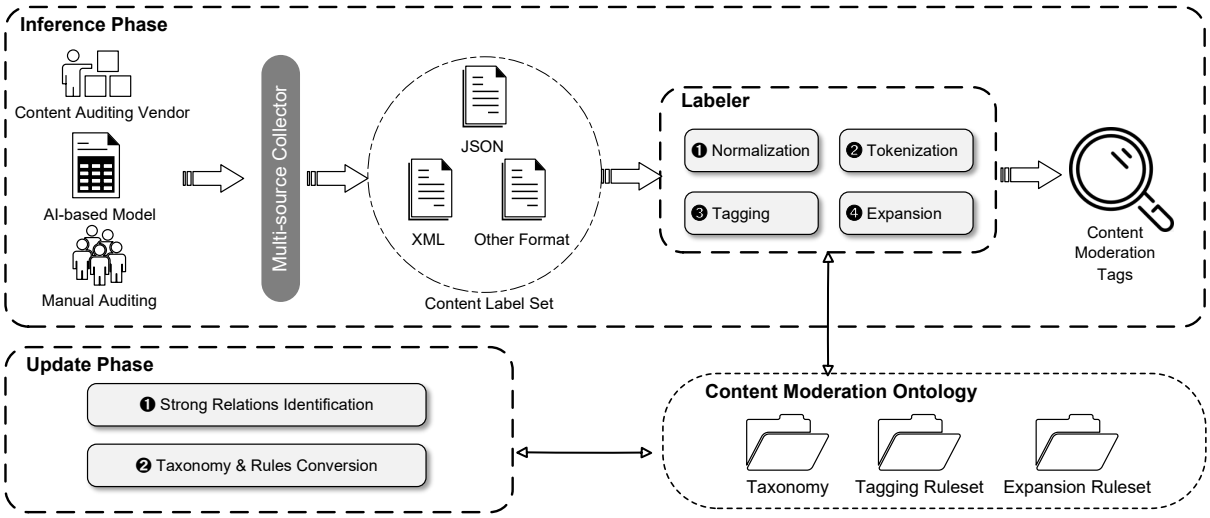

Figure 1: Overview of TCLens architecture.

diverse vendors or models will evolve in time to comply with regulatory requirements and technological advances [3, 10, 42]. Therefore, the ontology we built needs to be updated continuously. Marking unseen labels and concepts manually will seriously weaken the generalization capability of tags aggregation, and ultimately make the system unmaintainable. In view of this, TCLens provides an automatic update mechanism that enables the content moderation ontology including taxonomy, tagging rules, and expansion rules to conduct tag recognition better.

At a high-level perspective, TCLens consists of three parts: content moderation ontology, inference component, and update component. Figure 1 shows the framework of TCLens. The rest of this section will discuss these parts in detail.

3.2 Ontology

When a piece of AIGC is sent to a content auditing vendor to check its compliance, the vendor returns a vendor-specific audit result. To eliminate bias from different vendor data formats, we use the term *token* to map each meaningful label in the audit result, which will be described in detail in Section 3.3. These tokens usually contain AIGC auditing information such as content class including *violence, privacy, fraud, porn*, etc., keyword lists e.g. *alcohol, pistols, offensive signs*, and other useful information such as *multimodal types, recommended actions, multilingual identification*, etc [23, 40, 41]. To aggregate multi-source audit labels, we designed a content moderation ontology, including a taxonomy, tagging ruleset, and expansion ruleset, which respectively be leveraged in the knowledge base for tag aggregation, labels to tags conversion, and prediction from a known tag to previously unidentified tags. It is worth noting that the ontology we designed is open and can be updated manually or automatically. The ontology will be discussed in detail below:

3.2.1 ***Taxonomy.*** TCLens leverages a taxonomy as input and provides knowledge for tag inference, so the taxonomy is a structured tag set that organizes as a tree structure to represent the categories and relationships of content tags. Figure 2 shows a simplified taxonomy illustration, which includes a virtual root node (ROOT), and five sub-nodes representing categories including action (ACT), class (CLASS), keyword (KW), miscellaneous (MISC), and an important virtual node named unknown (UNK), which will be detailed in the rest of this section. It is worth mentioning that some categories do not contain intermediate nodes, such as the KW category, which only represents a set of content keywords, and the MISC category contains intermediate nodes e.g. LANGUAGE only used to denote the structure, as well as concrete tags, i.e. leaf nodes e.g. *English, Chinese*, etc.

We created a default taxonomy to introduce the TCLens workflow, consisting of five categories. However, since our approach features an open taxonomy that can be automatically refined during the update phase, this default taxonomy is simply an initial tool for setting up the knowledge base and does not significantly affect the overall effectiveness of TCLens.

- **Class (CLASS).** This category is used to mark the semantic classification of AIGC to be audited. For the content moderation mechanism, the CLASS mainly identifies non-compliant categories such as *violence, incivility, ban*, etc., and can be further divided into subcategories. According to the modal type of the content, for text content, NLP-based semantic recognition is used for tagging; for audio content, the classes are tagged by transcribing it into text [24]; for image content, the components in the image are identified through the AI model or manual labeling, and then the semantics of the image can be summarized [21]; for video content, several image frames within the video are extracted and converted into a series of images for recognition [38]. Additionally, generic tags such as *toxic* are discarded in the default taxonomy because those tags cannot distinguish between different content auditing results.
- **Keyword (KW).** The keyword collection of AIGC. They are used to represent key features of the content. We use a

flat structure instead of a hierarchical design in the default taxonomy, but parent-child relationships are still supported. In essence, the content's keywords are the concrete description of the corresponding CLASS tag. For example, if a piece of content contains the keyword *rifles*, its class tag may be *crime* or *terrorist*. For text, keywords are usually several words in the sentence or paragraph; for other modal content, keywords are key features of semantic recognition [39].

- **Miscellaneous (MISC).** This category reflects the characteristics of various aspects of AIGC and provides a crucial foundation for security analysts to make final decisions based on aggregated tags. For instance, it identifies multimedia streams to label the content's modality type for auditing [45]. In the case of text content, it detects the language used, which may include multiple languages within a single piece of content.
- **Action (ACT).** Each content moderation vendor provides an assessment of the content's compliance, determining the recommended action. In the default taxonomy, there are three action types. The *pass* tag indicates compliant content, while the *block* tag signifies content that requires blocking or further processing due to its toxic issues, e.g., data needing de-identification for privacy concerns. The *review* tag is applied when the content is flagged as abnormal, requiring manual inspection by security analysts. Since different moderation systems may assign different action labels to the same content, the final action tag is determined by the number of audit results for each label. If no action tag surpasses a predefined threshold, the tag defaults to *review*.
- **Unknown (UNK).** This is a virtual yet significant category. When a label in a piece of content cannot be tagged during the inference phase, it means that these labels cannot be effectively processed under the current ontology. Typically, there are two approaches to handle this: either discard the unrecognized labels or retain them for processing in later iterations after the update phase. In this paper, inspired by the AVclass2 [34] approach, we choose the latter method, categorizing these retained labels under the UNK category for further processing.

To build the default taxonomy, we manually reviewed real-world data from an AI enterprise and structured the taxonomy accordingly. Given its open design, other users can still create or refine their own taxonomy based on their specific datasets.

3.2.2 ***Tagging Ruleset.*** A set of relationships that maps content moderation labels (i.e., tokens) to one or more tags within the taxonomy described above. This tag mapping transforms the unstructured information from auditing labels into well-defined concepts. In our default ontology, there are four types of rules, outlined as follows:

- **Aliases.** Due to the absence of common standards, different content moderation vendors may assign different labels to the same content, even though they convey the same meaning. TCLens leverages the tagging ruleset to unify these labels during the inference phase, converting different labels with identical meanings into a single tag [11]. For example, the rude or abusive phrases "*what the hell*," "*what the fuck*," "*what the heck*," and "*wtf*" are often used to express the same sentiment, and we map them all to the tag "*wtf*".
- **Polysemy.** Polysemy, where a token in content auditing may implicitly convey multiple meanings, is common in natural language. To accurately capture the nuances of such tokens and ensure each tag represents the smallest meaningful unit, the tagging ruleset splits a polysemous token into multiple tags [12]. For example, the token "*swindle*" corresponds to two distinct class tags in our taxonomy: *fraud* and *crime*. This handling of polysemy allows TCLens to process aggregated content moderation tags at the most granular level.
- **Generic.** As far as we know, when a token is too generic to convey specific meaning, it should be discarded and not used as a tag to represent content characteristics, as it provides no valuable information for content moderation. Examples of such generic tokens include "*toxic*" and "*harmful*". If a token matches a generic tag in the tagging ruleset, TCLens discards it without additional processing.
- **Unknown.** In line with our design, TCLens does not aim to create a closed taxonomy that encompasses all content-related concepts, as we consider this an impractical task. Instead, when a token is not recognized in the tagging ruleset,

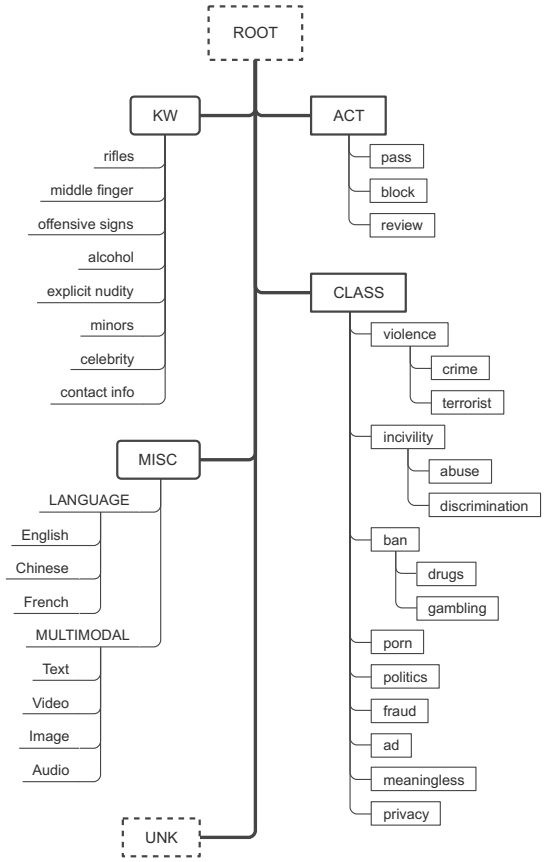

Figure 2: A simplified version of taxonomy with all five categories in the default taxonomy.

it is placed in the unknown category and assigned the UNK tag for further processing during the update phase.

TCLens offers a default tagging ruleset, which, like the taxonomy, is automatically updated as the system iterates and evolves with continued use.

3.2.3 ***Expansion Ruleset.*** Obviously, we can explicitly or implicitly infer one or more tags from a tag generated by the tagging ruleset mentioned before, and we refer to these inference rules as expansion ruleset. Inspired by the previous work AVclass [33, 34], we classify expansion rules into two types based on whether the inferred tags belong to the same category as the original tag.

- **Inter-Category Rule.** This rule type allows tags from one category to be inferred as tags in another category. For example, the tag *rifle* in the KW category can be inferred as tag *violence* in the CLASS category, indicating that content involving *rifles* is likely to be associated with *violence*. It is worth noting that an inter-category rule does not mean that AIGC must be definitively defined as the inferred tag. Instead, final tags are assigned based on the frequency threshold of tag occurrences across multiple sources of content moderation.
- **Intra-Category Rule.** Our taxonomy follows a hierarchical structure, where some categories have an inheritance relationship from the root node to its leaf nodes, forming an implicit expansion relationship known as an intra-category rule. For instance, in the taxonomy shown in Figure 2, content tagged with "*discrimination*" implicitly includes the "*incivility*" tag. Thus, when a node in the taxonomy has a meaningful parent node (represented by lowercase words in Figure 2), the child-parent node pair serves as an expansion rule.

In the implementation, we first apply inter-category expansion, followed by intra-category expansion, allowing cross-category inferred tags to have the chance to acquire their respective parent tags.

3.3 Inference Phase

In a typical operational model for generative AI services, when an AI model (e.g., an LLM) generates content based on user-submitted prompts, AIGC operators will call upon multiple content moderation vendors or classification models to obtain their review results. Generally, the input to the inference phase comprises multi-source content moderation results, and the output is a set of aggregated AIGC tags. During this phase, a multi-source auditing results collector stores review data in various formats, including JSON, XML, and plain text. Due to semantic misalignment across fields in these structured data, direct aggregation of labels from different audit sources is not feasible. For instance, the "*suggestion*" field in Tencent Cloud's text moderation system [9] indicates the recommended operation after reviewing the content, while in NetEase's moderation system [44], this corresponds semantically to the "*action*" field, yet each uses distinct enumerative values. In view of this, we designed a Labeler module, which performs four steps to aggregate semantically misaligned and non-uniformly formatted auditing results into a cohesive, high-confidence tag set.

Step 1: Normalization. For any data format of content auditing results, such as JSON or XML format, our insight is not to perform manual parsing for each moderation source. In fact, this approach is inefficient in the operation of generative AI services because, in real-world operations, different moderation sources continuously evolve their moderation capabilities, leading to changes in field formats and semantics. Therefore, for each moderation result, we extract meaningful information, such as values of *LanguageCode* and *Labels* fields in JSON and XML, and discard information unrelated to moderation labels, like *StatusCode* and *RequestId* fields. The retained labels are then merged into a list as the output.

Step 2: Tokenization. After normalizing the content moderation results across various formats, we further refine each item in the label list by filtering out non-printing characters, removing stop words, standardizing word separators, and adjusting letter case. Finally, each label list item is converted into a token for subsequent tagging procedure.

Step 3: Tagging. Next, each token in the label list is converted into a set of tags based on the tagging ruleset in the ontology discussed in Section 3.2.2. If a token matches an Aliases or Polysemy rule, it is converted into one or more tags accordingly. Tokens that hit the Generic rule are discarded, as they are likely meaningless to content moderation. Any tokens that do not fit those two cases are marked as Unknown type and reserved for further processing during the update phase. This step finally outputs a set of tags.

Step 4: Expansion. Lastly, the tags generated in the previous step, along with the expansion ruleset mentioned in Section 3.2.3, are used as input to create an expanded tag list. By applying inter-category rules, additional semantically inferred tags are added to the moderation results, while fine-grained tags are broadened to general tags through intra-category rules. This process establishes a robust foundation for producing a more comprehensive set of content moderation tags in the final output.

Finishing the above four steps, we obtain a list of tags and unknown tokens for an audit source. By applying this method iteratively, we generate tag lists and unknown tokens for each audit source. TCLens then counts the number of appearances of these tags and tokens, and only tags and tokens above the threshold (it is also can be configured) are retained. This frequency is used as the tag's confidence score. It is worth mentioning that although moderation vendors typically provide a confidence score [8, 9, 29, 44] , and AI models offer probability values [6, 26], these scores are determined based on each source's dataset and rules. They do not accurately reflect confidence across a more comprehensive, semantically aligned label set. Therefore, we ignore the moderation source-provided scores, instead relying on frequency counts from aggregated multi-source tags. The retained tags are then used as output to represent the feature of an AIGC item.

3.4 Update Phase

As TCLens accumulates more content auditing results, new content tags may emerge that are not in our ontology. Thus, a mechanism is required to update the ontology including taxonomy, tagging ruleset, and expansion ruleset. Using an automatic update strategy rather than a manual procedure, we apply a statistical approach similar to AVclass [34] to calculate the strong relationship between

tag pairs and update the ontology accordingly. This phase generally includes two steps, as detailed below:

Step 1: Strong Relations Identification.

Tags with high confidence in content moderation results generally meet two criteria: they appear frequently among all AIGC tags submitted for auditing and are consistently found in the feedback labels from different audit sources for the same AIGC. Based on this observation, we define and formalize the following concepts. Let $|t_i|$ and $|t_j|$ represent the frequency of tags t_i and t_j in all content moderation tags, while $|(t_i, t_j)|$ indicates the number of co-occurrence both t_i and t_j in moderation results. We then define $rel(t_i, t_j)$ in Equation 1 to express the ratio of co-occurrences of t_i and t_j in all tags containing t_i, measuring their relationship.

$$rel(t_i, t_j) = \frac{|(t_i, t_j)|}{|t_i|} \tag{1}$$

$$min(|t_i|, |t_j|) \geq n \tag{2}$$

$$rel(t_i, t_j) \geq \tau : t_i \Rightarrow t_j \tag{3}$$

$$rel(t_i, t_j) \geq \tau \wedge rel(t_j, t_i) \geq \tau : t_i \Leftrightarrow t_j \tag{4}$$

To quantify this relationship further, we define *strong relation*: a tag pair (t_i, t_j) that satisfies both Equation 2 and Equation 3 is considered a strong relation. For an even stricter condition, a tag pair is deemed "equivalent" if (t_i, t_j) meets both Equation 2 and Equation 4. In TCLens, the parameters n and τ are empirically selected, with recommended default values of $n = 8$ and $\tau = 0.8$.

Step 2: Taxonomy & Rules Conversion.

Given that a strong relation signifies a high likelihood of tag pairs co-occurring, we apply the following recursive process to update the taxonomy and ruleset:

(1) Input $SR = \{ sr \mid sr \text{ is strong relation} \}$, $sr = (t_i, t_j)$;

(2) Verify whether *sr* is **known**, meaning the relation is already defined in the current taxonomy, tagging ruleset, or expansion ruleset. If *sr* is identified, no action is needed, and the process skips directly to step (5). If not, proceed to the next step;

(3) If *sr* is **unknown** and **equivalent**, add the relation as a tagging rule;

(4) If *sr* is both **unknown** and **not equivalent**, further processing is carried out based on the categories of t_i and t_j. This step is configurable. For instance, if t_i is in the UNK category and t_j is in CLASS, a tagging rule from t_i to t_j is added; if t_i is in KW and t_j is in CLASS, an expansion rule from t_i to t_j is created;

(5) Remove *sr* and return to step (1), continuing until no relations remain to be processed in the iteration or until *SR* is empty.

Following these two steps, the ontology will be automatically updated based on the newly introduced dataset, enabling more effective aggregation of toxic tags.

4 EVALUATION

In this section, we present experiments evaluating the effectiveness of the proposed approach for AIGC tag aggregation. First, we outline the experimental setup and describe the audit sources used for evaluation. Then, we compare TCLens with other methods, examining precision and recall rates while exploring factors driving performance differences.

Table 1: AIGC Moderation Sources List.

ID	Moderation Source Name	Multimodal Type			
		Text	Audio	Image	Video
MS01	NEXTDATA Content Moderation Service [29]	✓	✓	✓	✓
MS02	Alibaba Cloud Content Moderation V2 [8]	✓	✓	✓	✓
MS03	Tencent Moderation System [9]	✓	✓	✓	✓
MS04	NetEase Yidun Content Security [44]	✓	×	✓	✓
MS05	LiveData Content Moderation [22]	✓	✓	✓	✓
MS06	TuringPlat Content Moderation [14]	✓	✓	×	×
MS07	ChatGLM V4 (fine tuning) [1]	✓	×	×	×
MS08	Stanford NLP [13]	✓	×	×	×
MS09	BERT [17]	✓	×	×	×
MS10	VGG [36]	×	×	×	✓

4.1 Experiment Setup

To evaluate the effectiveness of TCLens, we introduced several content moderation sources that provide audit results for submitted content through either service APIs or local function calls. Table 1 lists the selected moderation sources, comprising a total of 10 sources from two categories: external audit vendors and internally trained classification models. The self-built AI model was optimized for multimodal analysis to capture more comprehensive audit labels across different AIGC types. Using these sources, we developed a multi-source moderation collector and storage mechanism for further processing by the labeling module.

Following the methodology detailed in Section 3, we implemented a TCLens prototype. While our approach to AIGC tag aggregation draws inspiration from malware labeling techniques, a key distinction lies in how text-based AIGC content — comprising the largest segment - requires semantic similarity rather than strict equality for tag comparisons in the tagging and expansion processes. Unlike malware labels, text content comparison hinges on meaning rather than exact character matching. To accommodate this, we implemented text similarity techniques to align tagging and expansion rules, utilizing common NLP models i.e. *BERT model* and *siamese network* for tag similarity comparison.

4.2 Dataset

As aforementioned, the primary indicators for evaluating the effectiveness of aggregating massive harmful content labels are precision and recall. To collect extensive toxic AIGC labels, we constructed several datasets, summarized in Table 2. A total of six datasets were used in this evaluation, organized into two parts: The first part consists of four well-labeled datasets. Dataset D1, an open-source dataset provided by OpenAI [25], was used to evaluate detection rates and false positives across various categories. Datasets D2, D3, and D4 contain different types of toxic content generated in a controlled experiment environment and were manually labeled well. These four datasets serve as ground truth for evaluation, helping to reduce bias during the update phase. The second part comprises two real-world datasets sourced from two anonymous AIGC service providers. Content within these datasets was marked by at least

Table 2: Datasets List with Four of Ground Truth and Two of Real-world AIGC Services.

ID	Dataset	# pieces of AIGC
D1	OpenAI moderation evaluation dataset [25]	1680
D2	Well-labeled dataset on incivility, politics and violence	1018
D3	Well-labeled dataset on porn	473
D4	Well-labeled dataset on fraud, privacy and ad.	1615
D5	AIGC service A real-world data for 1 months	3674
D6	AIGC service B real-world data for 3 months	15822
	Total	**24282**

Table 3: Comparsion of Aggregation Effectiveness between TCLens and Baseline Methods.

Dataset ID	TCLens		Confidence Score		Manual Specified Source	
	Precision	Recall	Precision	Recall	Precision	Recall
D1	0.992	0.988	0.985	0.965	0.958	0.879
D2	0.996	0.992	0.988	0.935	0.966	0.894
D3	1.000	1.000	1.000	0.980	0.973	1.000
D4	0.962	0.938	0.960	0.917	0.977	0.812
D5	0.978	0.980	0.956	0.967	0.975	0.825
D6	0.985	0.992	0.945	0.975	0.948	0.867

one moderation source as requiring review or blocking, providing data to evaluate TCLens's aggregation effectiveness.

4.3 Aggregation Effectiveness

The critical metrics to evaluate the effectiveness of tag aggregation approaches for content moderation are the precision and recall of the inferred tags, so we conduct a comparison between TCLens with two baseline methods commonly used by AIGC services. The first approach sorts confidence scores within each audit source in descending order and retains tags with values above a threshold as AIGC tags. As shown in Table 3, both precision and recall are lower for this method compared to TCLens, and more importantly, the threshold selection (set to 0.8 in our experiment) heavily impacts label aggregation outcomes. The second method involves manually designating one moderation source as a high-weight source. However, this method significantly underperforms compared to TCLens due to the varying effectiveness of audit sources across different content types and datasets, making it challenging to select a single optimal source.

To verify the rationale behind our design by testing its functionality without the automatic update and expansion components, we conduct a ablation study for evaluating the recall rate of each components in TCLens. We compare the precision and recall rates of label aggregation results for each dataset, the experiment results show that removing the automatic update and expansion components almost have no impact on precision rate. This aligns with our intuition that update and expansion processes would introduce more effective labels, but could hardly prune redundant labels. However, those two processes could the recall rate as shown in Figure 3. Experimental results indicate that TCLens generally does not perform well when either the update phase or the expansion procedure is removed, except for dataset D4. A manual review of the D4 results revealed that this is due to the predominance of fraud and advertisement data in the dataset. Excessive inference tags in these categories introduce a higher rate of false positives, highlighting a direction for future research to improve precision in similar cases.

Additionally, we observed that, while the performance of different methods varies, there is a consistent trend in precision and recall rates across datasets. Our default ontology was developed using well-labeled datasets D1, D2, D3, and D4, which leads to better performance across all methods on these datasets compared to the other two. Although TCLens also shows strong results on D5 and D6, the generalization ability of our approach also needs to be further improved. Moreover, the experiments indicate that datasets with a single content type, such as D3, yield better results than composite datasets, suggesting that label aggregation is more effective for AIGC services focused on specific domains.

5 CONCLUSION

In this paper, we propose TCLens, a toxicity tags aggregation-based approach to content moderation for AIGC. This method presents an ontology of content moderation including taxonomy, tagging ruleset, and expansion ruleset, and leverages an automatic inference and update framework to aggregate content auditing results and optimize the ontology. The systematic evaluation shows that our approach has excellent precision and recall rate in real-world AIGC service and outperforms existing SOTA solutions.

Admittedly, as AIGC technology advances, including developments in hyper-anthropomorphism, emerging content security challenges may arise that fall beyond our current observable paradigm. Addressing these challenges will be a focus of our future research.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions. This work is partially supported by Beijing Computer Federation (BCF) and Intelligent Security Collaborative Innovation Group of ISCAS-AHUT.

REFERENCES

[1] ZhiPu AI. 2024. ChatGLM V4. https://chatglm.cn.
[2] Arslan Akram. 2023. An empirical study of ai generated text detection tools. *arXiv preprint arXiv:2310.01423* (2023).
[3] Naeem AllahRakha. 2023. AI and the law: Unraveling the complexities of regulatory frameworks in Europe. *International Bulletin of Young Scientist* 1, 2 (2023).
[4] Baran Barbarestani, Isa Maks, and Piek TJM Vossen. 2024. Content Moderation in Online Platforms: A Study of Annotation Methods for Inappropriate Language. In *Proceedings of the Fourth Workshop on Threat, Aggression & Cyberbullying@ LREC-COLING-2024*. 96–104.
[5] Nailya I Batrova, Lyajlya L Salekhova, Guler Cavusoglu, and Marina A Lukoyanova. 2017. Designing of content of the bilingual elective course "Information and communication technologies (Ict)". *Modern Journal of Language Teaching Methods(MJLTM)* 7, 9.1 (2017), 127–130.
[6] Nadia Burkart and Marco F Huber. 2021. A survey on the explainability of supervised machine learning. *Journal of Artificial Intelligence Research* 70 (2021), 245–317.
[7] Chuan Chen, Zhenpeng Wu, Yanyi Lai, Wenlin Ou, Tianchi Liao, and Zibin Zheng. 2023. Challenges and remedies to privacy and security in aigc: Exploring the potential of privacy computing, blockchain, and beyond. *arXiv preprint arXiv:2306.00419* (2023).
[8] Alibaba Cloud. 2023. Content Moderation 2.0. https://www.alibabacloud.com/help/en/content-moderation/latest/content-security-plus.
[9] Tencent Cloud. 2022. Text Moderation System API Documentation. https://main.qcloudimg.com/raw/document/intl/product/pdf/tencent-cloud_1121_43775_en.pdf.

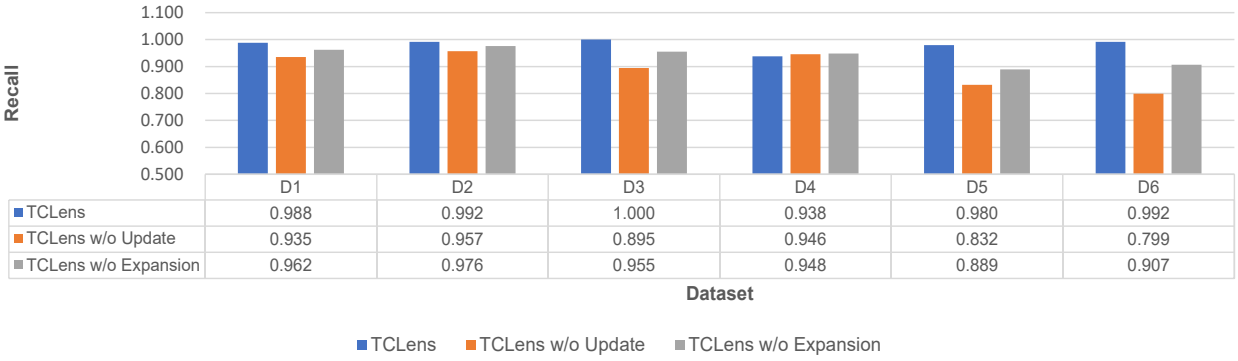

Figure 3: Ablation study of TCLens, w/o update phase and w/o expansion procedure.

[10] Esther Franks, Bianca Lee, and Hui Xu. 2024. Report: China's New AI Regulations. *Global Privacy Law Review* 5, 1 (2024).

[11] Anna D Gibson, Niall Docherty, and Tarleton Gillespie. 2024. Health and toxicity in content moderation: The discursive work of justification. *Information, Communication & Society* 27, 7 (2024), 1441–1457.

[12] Sebastiaan Gorissen. 2024. Weathering and weaponizing the# TwitterPurge: digital content moderation and the dimensions of deplatforming. *Communication and Democracy* 58, 1 (2024), 1–26.

[13] The Stanford Natural Language Processing Group. 2020. Classifier. https://nlp.stanford.edu/.

[14] iFLYTEK. 2024. TuringPlat AI. http://www.turingplat.com/index.

[15] Yongkang Jiang, Gaolei Li, and Shenghong Li. 2023. Tagclass: a tool for extracting class-determined tags from massive malware labels via incremental parsing. In *2023 53rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. IEEE, 193–200.

[16] K Karthikeyan and V Karthikeyani. 2015. Ontology based concept hierarchy extraction of web data. *Indian Journal of Science and Technology* 8, 6 (2015), 536.

[17] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, Vol. 1. Minneapolis, Minnesota, 2.

[18] Sunder Ali Khowaja, Parus Khuwaja, Kapal Dev, Weizheng Wang, and Lewis Nkenyereye. 2024. Chatgpt needs spade (sustainability, privacy, digital divide, and ethics) evaluation: A review. *Cognitive Computation* (2024), 1–23.

[19] Mahi Kolla, Siddharth Salunkhe, Eshwar Chandrasekharan, and Koustuv Saha. 2024. LLM-Mod: Can Large Language Models Assist Content Moderation?. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*. 1–8.

[20] Deepak Kumar, Yousef Anees AbuHashem, and Zakir Durumeric. 2024. Watch Your Language: Investigating Content Moderation with Large Language Models. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 18. 865–878.

[21] Yinglong Li. 2022. Research and application of deep learning in image recognition. In *2022 IEEE 2nd international conference on power, electronics and computer applications (ICPECA)*. IEEE, 994–999.

[22] LiveData. 2024. Content Moderation. https://www.ilivedata.com/.

[23] Lingjuan Lyu, C Chen, and J Fu. 2023. A Pathway Towards Responsible AI Generated Content.. In *IJCAI*. 7033–7038.

[24] Mishaim Malik, Muhammad Kamran Malik, Khawar Mehmood, and Imran Makhdoom. 2021. Automatic speech recognition: a survey. *Multimedia Tools and Applications* 80 (2021), 9411–9457.

[25] Todor Markov, Chong Zhang, Sandhini Agarwal, Tyna Eloundou, Teddy Lee, Steven Adler, Angela Jiang, and Lilian Weng. 2022. A Holistic Approach to Undesired Content Detection. *arXiv preprint arXiv:2208.03274* (2022).

[26] Joao Marques-Silva. 2023. Logic-based explainability in machine learning. In *Reasoning Web. Causality, Explanations and Declarative Knowledge: 18th International Summer School 2022, Berlin, Germany, September 27–30, 2022, Tutorial Lectures*. Springer, 24–104.

[27] Midjourney. 2024. AI. https://www.midjourney.com/home.

[28] Sankha Subhra Mullick, Mohan Bhambhani, Suhit Sinha, Akshat Mathur, Somya Gupta, and Jidnya Shah. 2023. Content moderation for evolving policies using binary question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*. 561–573.

[29] NEXTDATA. 2024. Content Moderation API Documentation. https://intl.ishumei.com/help/api.

[30] OpenAI. 2024. ChatGPT. https://openai.com/chatgpt/overview/.

[31] OpenAI. 2024. DALL-E 3. https://openai.com/index/dall-e-3/.

[32] Rohan Singh Rajput, Sarthik Shah, and Shantanu Neema. 2023. Content moderation framework for the LLM-based recommendation systems. *Journal of Computer Engineering and Technology (IJCET)*. 14, 3 (2023), 104–17.

[33] Marcos Sebastián, Richard Rivera, Platon Kotzias, and Juan Caballero. 2016. Avclass: A tool for massive malware labeling. In *Research in Attacks, Intrusions, and Defenses: 19th International Symposium, RAID 2016, Paris, France, September 19-21, 2016, Proceedings 19*. Springer, 230–253.

[34] Silvia Sebastián and Juan Caballero. 2020. Avclass2: Massive malware tag extraction from av labels. In *Proceedings of the 36th Annual Computer Security Applications Conference*. 42–53.

[35] Changho Shin and Alice Schoenauer Sebag. [n. d.]. Can we get smarter than majority vote? Efficient use of individual rater's labels for content moderation. ([n. d.]).

[36] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[37] Statista. 2023. Research AI. https://www.statista.com/research-ai/.

[38] Hao Tang, Lei Ding, Songsong Wu, Bin Ren, Nicu Sebe, and Paolo Rota. 2023. Deep unsupervised key frame extraction for efficient video classification. *ACM Transactions on Multimedia Computing, Communications and Applications* 19, 3 (2023), 1–17.

[39] Swasthika Jain Thandaga Jwalanaiah, Israel Jeena Jacob, and Ajay Kumar Mandava. 2023. Effective deep learning based multimodal sentiment analysis from unstructured big data. *Expert Systems* 40, 1 (2023), e13096.

[40] Taoye Wang, Li Li, Xiang Chen, and Kunzhu Li. 2024. A Study on the Risks and Countermeasures of False Information Caused by AIGC. *Journal of Electrical Systems* 20, 3 (2024), 420–426.

[41] Tao Wang, Yushu Zhang, Shuren Qi, Ruoyu Zhao, Zhihua Xia, and Jian Weng. 2023. Security and privacy on generative data in aigc: A survey. *arXiv preprint arXiv:2309.09435* (2023).

[42] Yuntao Wang, Yanghe Pan, Miao Yan, Zhou Su, and Tom H Luan. 2023. A survey on ChatGPT: AI-generated contents, challenges, and solutions. *IEEE Open Journal of the Computer Society* (2023).

[43] Jiayang Wu, Wensheng Gan, Zefeng Chen, Shicheng Wan, and Hong Lin. 2023. Ai-generated content (aigc): A survey. *arXiv preprint arXiv:2304.06632* (2023).

[44] NetEase Yidun. 2024. Content Security. https://dun.163.com/locale/en/content-security.

[45] Tianlun Zheng, Zhineng Chen, Bingchen Huang, Wei Zhang, and Yu-Gang Jiang. 2023. Mrn: Multiplexed routing network for incremental multilingual text recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 18644–18653.

[46] Wanzheng Zhu, Hongyu Gong, Rohan Bansal, Zachary Weinberg, Nicolas Christin, Giulia Fanti, and Suma Bhat. 2021. Self-supervised euphemism detection and identification for content moderation. In *2021 IEEE Symposium on Security and Privacy (SP)*. IEEE, 229–246.