Struck-Out Word Detection in Handwritten Text: A Cross-Dataset Evaluation on the HWG Dataset

Said Yasin[1], Christian Gold[2], Torsten Zesch[1]
[1]CATALPA – Center of Advanced Technology for Assisted Learning and Predictive Analytics, FernUniversität in Hagen, Germany
[2]Universität Duisburg-Essen, Forsthausweg 2, 47057 Duisburg, Germany
Correspondence: torsten.zesch@fernuni-hagen.de

Abstract

Detecting struck-out words in handwritten text is important for downstream applications such as educational assessment, where struck-out content should not be included in the final transcript. However, progress on this task is limited by the scarcity of publicly available datasets and by the lack of systematic evaluations across datasets and strike-out types. In this paper, we introduce several datasets with consistent annotations for multiple strike-out types and provide an open-source evaluation setup for word-level struck-out detection. We show that strong intra-dataset performance does not necessarily translate into robust cross-dataset generalization, and that our newly created data improves generalization, especially for rare strike-out types.

1 Introduction

Detecting struck-out words in handwritten text is an important task especially in educational settings, as struck-out words might still be recognized (Nisa et al., 2019; Likforman-Sulem and Vinciarelli, 2008) and included in the final transcript. This may negatively impact downstream applications, such as the automated scoring of handwritten exams (Gold and Zesch, 2020a), where words that a student has crossed out should not be considered during grading. Figure 1 shows an example answer to the question "Will iron rust if you keep it completely dry?" in which retaining the struck-out negation reverses the intended meaning and hence the score to be awarded.

Therefore, in this work, we focus on classifying handwritten words as 'struck-out' or 'non-struck-out' (Shivakumara et al., 2021; Poddar et al., 2021) in order to remove the detected struck-out words before handwritten text recognition.[1]

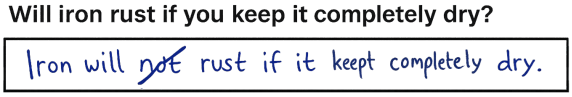


Figure 1: Example sentence with a struck-out word changing the intended meaning.

A major bottleneck for such classifiers is that datasets with handwritten struck-out words remain scarce (Nigam et al., 2024) and contain only a small number of struck-out words, which limits systematic research on this topic (Dheemanth Urs and Chethan, 2021). Larger existing resources are synthetic (Nisa, 2023; Heil et al., 2021a) or not publicly available (Brink et al., 2008; Chaudhuri and Adak, 2017). Research conducted on such data (Nigam et al., 2023; Shivakumara et al., 2022; Zhong et al., 2025; Chaudhuri and Adak, 2017; Brink et al., 2008) is hard to replicate and it remains difficult to assess potential generalization to real handwritten data. To address these limitations, we contribute a new dataset with struck-out words, consistent strike-out type annotations across multiple datasets, and an open-source setup for cross-dataset and struck-out type evaluations.

Using this setup, we show with both a CNN-based YOLO classifier and a transformer-based DINO classifier that previous work likely overestimates struck-out word classification performance, as results drop substantially in a broad cross-dataset

[1]This is in contrast to another line of research that aims to remove strike-out markings to improve recognition accuracy (Heil et al., 2021b; Bhattacharya et al., 2017; Chaudhuri and Adak, 2017).

evaluation despite strong intra-dataset performance. We further show that our newly created data improves generalization, especially for rare strike-out types, while additional annotated struck-out words would likely provide only limited further gains, since the learning curves saturate early. We make our code and instructions for using our datasets publicly available at https://github.com/said702/hwg-strikeout-detection.

2 Existing Approaches and Datasets

Table 1 gives an overview of previous studies and the obtained results in terms of F_1 for detecting struck-out words in a binary classification setting. Several studies report promising results on real data, with F_1 scores above .85. However, these results are difficult to compare, since most approaches are evaluated only in single-dataset settings and therefore mainly reflect intra-dataset performance rather than robust cross-dataset generalization. The same limitation also applies to mixed settings that combine synthetic and real data: although Shivakumara et al. (2021) and Zhong et al. (2025) report high F_1 scores of .94, these results provide only limited evidence of generalization to purely real, independent target data. For synthetic-to-real transfer, the overall picture remains unclear. While Heil et al. (2021b) report that performance drops from .99 on synthetic test data to .78 on real data, indicating a domain gap, Poddar et al. (2021) achieve an F_1 score of .93 when training on synthetic data and evaluating on real data, suggesting that synthetic training data can still be beneficial. Overall, the empirical basis is limited as there is a lack of comparable results on publicly available datasets, especially for cross-dataset setups. It is also largely unknown how performance varies across strike-out types. Moreover, previous work focuses mainly on the struck-out class, although the non-struck-out class is particularly important in educational settings, where false positives may remove words from the final transcript and distort students' intended meaning.

2.1 Synthetic Data

The upper part of Table 2 gives an overview of existing synthetic datasets. As they are relatively easy to construct, they tend to be publicly available and mainly differ in how the strike-out patterns are overlaid on the word samples. For example, in Poddar et al. (2021), the strike-out patterns were handwritten separately and then synthetically overlaid onto IAM word images, whereas Nigam et al. (2023) and Nisa (2023) generated the strike-out patterns directly based on image statistics. While such methods enable broad coverage of strike-out types, they may generalize less well to out-of-distribution data with different visual characteristics.

2.2 Real Data

The lower part of Table 2 gives an overview of existing datasets with real struck-out samples. Overall, such publicly available data are surprisingly scarce. We are only aware of one such case, the Single-Writer Strikethrough (SWS) dataset[2] by Heil et al. (2021b). The SOW dataset (Zhong et al., 2025) can be obtained upon request, which is theoretically also true for the data from Adak et al. (2017), but at the time of writing, we did not receive an answer. Over time, 'available upon request' often degrades into 'not available at all', which makes truly public resources even more important. We thus create public datasets, described in more detail in the next section.

3 The ~~HWG~~ Dataset

To address the limited availability of suitable annotated samples for struck-out word detection, we introduce the ~~HWG~~ dataset. It consists of three complementary subsets: *collected*, *written*, and *synthetic* as well as a strike-out type annotation of the *SOW* dataset. Details on availability, licensing, and source-specific redistribution restrictions are provided in Section 3.6.

3.1 Definition of Strike-out Types

Previous work has categorized strike-out marks into several recurring forms. Adak et al. (2017) distinguish six broad types: *single*, *multiple*, *slanted*, *crossed*, *zigzag*, and *wavy*. Heil et al. (2021b) extend this inventory to seven types by adding *scratch* as a further category. Building on these earlier categorizations, we define a more fine-grained set of nine strike-out types by splitting the broad *single* and *multiple* categories into *horizontal* and *oblique* variants, treating the earlier *slanted* category as part of the *oblique* variants, and additionally introducing the new types *circled* and *blackened*.[3] The resulting nine strike-out types are *single-horizontal*, *single-oblique*,

[2] https://zenodo.org/records/4765063

[3] The *blackened* category is conceptually close to the *scratch* type introduced by Heil et al. (2021b).

Reference	Method	Train	Test	F_1
Brink et al. (2008)	Connected components	-	real	$< .64$[a]
Adak and Chaudhuri (2014)	Connected components	-	real	.87
Chaudhuri and Adak (2017)	SVM	real	real	.92
Nisa et al. (2019)	CRNN + BiLSTM	synthetic	synthetic	.83[b]
Heil et al. (2021b)	DenseNet121	synthetic	synthetic real	.99 .78
Shivakumara et al. (2021)	DenseNet121	mixed	mixed	.94
Poddar et al. (2021)	SVM	synthetic	real	.93
Zhong et al. (2025)	ResNet50	mixed	synthetic mixed[c]	.84 .94

[a] Since the paper does not report an F_1 score, we estimated the maximum possible value from the reported true positive rate of 47.5% and true negative rate of 99.1%, assuming a best-case precision of 100%.
[b] F_1 score not reported explicitly, but could be derived from the given confusion matrix.
[c] We categorize this as 'mixed' despite the paper's claim to use real data, since qualitative inspection indicates the test set includes both real and synthetic samples.

Table 1: Overview of prior approaches for struck-out handwritten text detection.

multiple-horizontal, *multiple-oblique*, *crossed*, *circled*, *wavy*, *zigzag*, and *blackened*. Representative examples are shown in Table 3.

3.2 ~~HWG~~-collected

Struck-out words occur naturally in handwritten documents when writers revise or correct their text. Such examples are therefore present, although only in limited numbers, in several existing handwriting datasets. This is the case for the word-based GoBo dataset (Gold et al., 2021), in which struck-out words are marked with the label 'cr' in the ground truth. Similarly, the word-based version of the IAM dataset (Marti and Bunke, 2002) contains struck-out words annotated with the symbol '#'. In addition, document-level datasets such as the handwritten ASAP SAS dataset (Gold and Zesch, 2020b) contain struck-out words in handwritten student answer sheets, although these are not explicitly represented in the ground truth.

To turn these naturally occurring examples into a usable dataset, we collected struck-out samples from these datasets. This was straightforward for IAM and GoBo, since both are word-based datasets and explicitly indicate struck-out words in the ground truth. In contrast, ASAP required substantially more manual effort, as struck-out word instances first had to be manually identified and extracted from the document images.

Annotating Strike-out Types After collecting all samples from the three datasets, we annotated the struck-out words according to the strike-out types introduced in Section 3.1. The annotation was performed by two independent annotators. Inter-annotator agreement in terms of Cohen's κ was .66 for all items and .83 for the roughly 75% of items for which annotators were certain about their decisions. All disagreements were adjudicated by a third annotator. To additionally include non-struck-out samples, we sampled as many non-struck-out words from the GoBo, IAM, and ASAP datasets as struck-out samples had been collected. Table 4 summarizes the frequencies of all strike-out types and the non-struck-out samples.

3.3 ~~HWG~~-SOW

To enable cross-dataset evaluation, we additionally included the SOW dataset (Zhong et al., 2025), introduced in Section 2.2. The dataset contains struck-out words extracted from 66 handwritten student answer sheets and was obtained from the authors. Before annotating the strike-out types, we excluded synthetic struck-out samples, leaving 295 real struck-out samples.

Annotating Strike-out Types As the SOW dataset provides only binary labels indicating whether a word is struck out or not, two annotators assigned the same nine strike-out types as those used in ~~HWG~~-collected.[4] In case of disagreement, the two annotators collaboratively reviewed the conflicting cases and resolved them by mutual agreement following discussion. The distribution of the types (see Table 4) shows that the types are unevenly distributed in the collected data, likely

[4] Only eight were actually observed in the dataset.

Type	Name	Reference	# ~~Words~~	# Styles	Availability
Synthetic	-	Poddar et al. (2021)	103,901	6	closed
	-	Nigam et al. (2023)	n/a	7	upon request
	IAM Strikethrough	Heil et al. (2021a)	4,158	7	public [a]
	IAM synthetic	Nisa (2023)	2,996	4	public
	~~HWG~~-synthetic	this paper	37,010	7	public
Real	-	Brink et al. (2008)	624	n/a	closed
	-	Adak and Chaudhuri (2014)	385	n/a	closed [b]
	-	Chaudhuri and Adak (2017)	2,827	6	closed [c]
	-	Poddar et al. (2021)	443	6	closed
	SOW	Zhong et al. (2025)	648	5	upon request
	~~HWG~~-SOW	this paper	295	8	labels only
	SWS	Heil et al. (2021b)	630	7	public
	~~HWG~~-collected	this paper	402	9	public
	~~HWG~~-written	this paper	1,848	9	public

[a] Although the published dataset contains only 4,158 struck-out samples, the released generation method can also be used to create larger synthetic datasets, such as ~~HWG~~-synthetic in this paper.
[b] This dataset may be an earlier version of the dataset used in Chaudhuri and Adak (2017).
[c] The dataset is reported as being available upon request. However, we were not able to obtain access during the preparation of this work.

Table 2: Overview of datasets with real and synthetic struck-out samples.

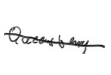
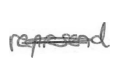
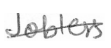
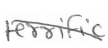
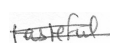
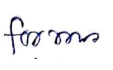
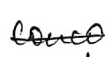
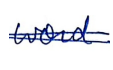
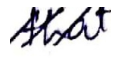
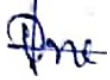
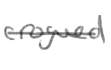
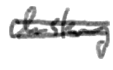
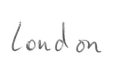
	Non struck-out	Single-horizontal	Single-oblique	Multiple-horizontal	Multiple-oblique	Crossed	Circled	Wavy	Zigzag	Blackened
~~HWG~~-collected										
~~HWG~~-written										
~~HWG~~-SOW									-	
~~HWG~~-Synthetic					-		-			
SWS					-		-			

Table 3: Annotated examples of strike-out types across all datasets.

reflecting writer-specific or culture-specific preferences. For example, *single-oblique* is the most frequent type in ~~HWG~~-collected but is rare in SOW, whereas the opposite holds for *circled*. The type *zigzag* does not appear at all in SOW.

3.4 ~~HWG~~-written

Some strike-out types (*multiple-oblique*, *wavy*, and *zigzag*) are so rare in the ~~HWG~~-collected and ~~HWG~~-SOW datasets that the empirical basis remains insufficient. Therefore, we conducted a targeted data collection study to achieve broader and more balanced coverage across strike-out types. A total of 17 writers participated, including 9 male and 8 female writers. As strike-out styles may be influenced by early writing instruction, we included participants with diverse educational and cultural backgrounds, including Chinese, German, Indian, Japanese, and Russian backgrounds. Each participant was instructed to write 12 words for each strike-out type. We refer to this dataset as ~~HWG~~-written.

Annotating Strike-out Types In contrast to the collected datasets, retrospective type annotation was not necessary for ~~HWG~~-written, since the strike-out types were predefined during data collection through an instruction sheet with illustrative examples (see Figure 8 in Appendix A).

Type	Real-collected		Real-written		Synthetic
	~~HWG~~ collected	~~HWG~~ SOW	~~HWG~~ written	SWS	~~HWG~~ synthetic
Total	804	4,826	2,052	756	74,020
Non-struck-out	402	4,531	204	126	37,010
Single-horizontal	74	96	204	90	5,281
Multiple-horizontal	34	50	204	90	5,341
Single-oblique	105	11	204	90	5,321
Multiple-oblique	2	17	216[a]	–	–
Crossed	69	2	204	90	5,356
Circled	8	104	204	–	–
Wavy	3	10	204	90	5,230
Zigzag	10	0	204	90	5,230
Blackened	97	5	204	90	5,251
Total struck-out	402	295	1,848	630	37,010
Struck-out ratio	.50	.06	.90	.83	.50

[a] The count of 216 includes 12 additional Multiple-oblique samples from writer 6 that were retained.

Table 4: Distribution of strike-out types in annotated datasets.

3.5 ~~HWG~~-synthetic

Since previous work has shown that synthetically generated strike-out patterns may help detect struck-out words (Zhong et al., 2025; Poddar et al., 2021), we additionally generated a synthetic dataset. We apply the method by Heil et al. (2021b) that derives image-dependent statistics from each input word image, such as stroke width, core region, and ink intensity, to generate a synthetic strike-out stroke that is then deformed and blended with the original image. Applied to the full GoBo dataset, this resulted in 37,010 synthetically struck-out samples across seven strike-out types.[5] Combined with the 37,010 original samples, the full ~~HWG~~-synthetic dataset comprises 74,020 samples.

3.6 Data Availability and Licensing

The ~~HWG~~ Dataset is publicly available on Zenodo (Gold et al., 2026). The release includes the newly collected samples and all redistributable data derived from ASAP and GoBo. For IAM and SOW, only identifiers and annotations are provided; the original images must be obtained from the respective source datasets. Users must comply with the applicable licenses, access conditions, and citation requirements.

4 Experimental Setup

In this section, we describe the experimental setup used to train and evaluate models for struck-out word detection.

[5] Following Heil et al. (2021b), the synthetic subset covers only seven strike-out types and therefore does not include the *circled* and *multiple-oblique* types.

Dataset	Median size
~~HWG~~-collected	164×77 px
~~HWG~~-SOW	134×66 px
~~HWG~~-written	195×72 px
SWS	319×220 px
~~HWG~~-synthetic	173×69 px

Table 5: Median image dimensions of the word-level crops.

Metrics We formulate strike-out detection as a binary word-level classification task (*struck-out* vs. *non-struck-out*). To evaluate system performance, we use the standard metrics *precision*, *recall*, and *F1-score*.

Models We train two models for struck-out word classification: YOLO and DINO.

From the **YOLO** family we employ the most recent classification variant `YOLO26n-cls`[6]. YOLO26 performs image classification by resizing and normalizing the input image, extracting multi-scale features with a convolutional backbone, aggregating them in a lightweight neck, and applying a classification head for single-label prediction (Sapkota et al., 2026).

Additionally, we use a Vision Transformer (Dosovitskiy et al., 2021) whose backbone is initialized with self-supervised pretrained weights from **DINO** v2 (Oquab et al., 2024). The model performs image classification by splitting the input image into fixed-size patches, converting them into embeddings with positional information, and processing them with self-attention-based Transformer encoder layers.

In addition, we evaluate recent vision-language models (VLMs) in a zero-shot setting.

Data For our experiments, we use the datasets we created: ~~HWG~~-collected, ~~HWG~~-SOW, ~~HWG~~-written, and ~~HWG~~-synthetic. In addition, we use the public SWS dataset introduced in Section 2.2. All datasets used in our experiments consist of word-level image crops. Table 5 reports the median original crop dimensions.

Data Splits In intra-dataset evaluation, we use 10-fold cross-validation: in each fold, 10% is held out for testing, and the remaining 90% is split into training and validation data, resulting in effective

[6] `https://docs.ultralytics.com/tasks/classify/#models`

Model	$F_1(s)$		$F_1(\neg s)$	
	Intra	Cross	Intra	Cross
YOLO	.96	.80	.97	.75
DINO	.93	.77	.92	.73

Table 6: Macro-averaged intra- and cross-dataset F_1 scores for struck-out (s) and non-struck-out ($\neg s$).

splits of 72%, 18%, and 10%. Results are averaged across folds. In cross-dataset evaluation, the source dataset is split into 80% training and 20% validation data, and the model is tested on the complete target dataset.

Training Configuration Across all experiments, each model was trained for 40 epochs, with a batch size of 32 for YOLO26 and 8 for DINOv2, due to the transformer architecture's higher memory requirements. For both models, class weighting was applied in the loss function to compensate for class imbalance when either struck-out or non-struck-out samples were overrepresented in a dataset.

5 Results

This section presents the experimental results of the systems for detecting struck-out words across the different datasets.

5.1 Cross-Dataset Performance

Since prior work has mostly reported struck-out classification results on individual datasets (Chaudhuri and Adak, 2017; Shivakumara et al., 2021; Zhong et al., 2025), it remains unclear to what extent strong intra-dataset performance translates into robust cross-dataset generalization. We therefore begin with an aggregated comparison of intra-dataset and cross-dataset performance across the three real-data subsets of ~~HWG~~ and the SWS dataset (see Table 6). For both architectures and both classes, intra-dataset performance is high, but cross-dataset performance is consistently lower. For the struck-out class, both models show a drop of .16 in F_1, while the non-struck-out class drops by .22 for YOLO and .19 for DINO. These results indicate that evaluations on individual datasets probably overestimate the real-world robustness of strike-out detection models.

Dataset-Specific Analysis To better understand the aggregated results, we first break them down at the dataset level for the struck-out class. Table 7 reports the resulting F_1 scores for this class.[7] For comparison, we also report intra-dataset results on the main diagonal, which are generally high (0.89–1.00 for YOLO and 0.83–1.00 for DINO). This is in line with the results from the literature shown in Table 1. The off-diagonal entries show that the performance drop is not equally strong for all dataset pairs, but is particularly large for some transfer settings. For instance, YOLO drops from 0.99 on ~~HWG~~-written to 0.33 when evaluated on ~~HWG~~-SOW, mainly due to low precision (0.20) despite high recall (0.95), indicating that the model over-predicts the struck-out class. When comparing YOLO and DINO across training datasets, YOLO performs better on average for ~~HWG~~-collected, ~~HWG~~-SOW, and SWS, whereas DINO performs better for ~~HWG~~-written and especially ~~HWG~~-synthetic. Since YOLO performs better on more datasets overall and yields stronger intra- vs. cross-dataset results, we report only YOLO results in the following experiments.

Training dataset	Model	Evaluation dataset				
		~~HWG~~-collected	~~HWG~~-SOW	~~HWG~~-written	~~HWG~~-synthetic	SWS
~~HWG~~-collected	YOLO	(0.97)	.77	.91	.87	.81
	DINO	(0.94)	.69	.88	.86	.72
~~HWG~~-SOW	YOLO	.79	(0.89)	.85	.86	.78
	DINO	.86	(0.83)	.80	.74	.70
~~HWG~~-written	YOLO	.89	.33	(0.99)	.85	.98
	DINO	.86	.45	(0.98)	.86	.95
~~HWG~~-synthetic	YOLO	.08	.00	.07	(1.00)	.21
	DINO	.78	.51	.89	(1.00)	.77
SWS	YOLO	.83	.68	.96	.80	(1.00)
	DINO	.90	.43	.96	.82	(0.98)

Table 7: F_1 scores for the struck-out class.

5.2 Influence of Strike-out Type

While the previous results showed overall cross-dataset performance, they did not indicate how well the models generalize to specific strike-out types. This can now be assessed in a more fine-grained manner because we annotated all strike-out types. Figure 2 shows cross-dataset F_1 scores by strike-out type for YOLO trained on ~~HWG~~-written

[7] The corresponding precision and recall scores are reported in Table 11 in Appendix B.

or ~~HWG~~-collected and evaluated on the remaining datasets.[8] The results suggest that ~~HWG~~-written is more effective for rare strike-out types. This is most evident on SWS, which contains an equal number of samples per type. For *wavy* and *zigzag*, the model trained on ~~HWG~~-written clearly outperforms the one trained on ~~HWG~~-collected, reaching F_1 scores of 0.93 and 0.94 compared with 0.12 and 0.00. This likely reflects the much larger number of *wavy* and *zigzag* samples available for training in ~~HWG~~-written than in ~~HWG~~-collected (204 vs. 3 and 204 vs. 10; see Table 4). At the same time, YOLO trained on ~~HWG~~-collected performs better in some settings, such as *crossed* (1.00 vs. 0.80) and *blackened* (0.89 vs. 0.73) on ~~HWG~~-SOW, possibly because this dataset consists of naturally occurring strike-out markings despite its smaller size. Figure 3 shows misclassified *blackened* samples on ~~HWG~~-SOW for the model trained on ~~HWG~~-written, indicating that the model is not yet fully robust even for visually clear cases.

Unseen Strike-out Types Another relevant question is how the classifier handles a new strike-out type not represented during training. To investigate this, we conducted a leave-one-type-out experiment: a YOLO classifier was trained on all strike-out types from the ~~HWG~~-written dataset except one, and the held-out type was evaluated together with an equal number of non-struck-out samples from ~~HWG~~-collected. This was repeated for each strike-out type. Across all held-out classes, the F_1 scores ranged between 0.88 and 0.91, suggesting that the classifier captures general visual properties of strike-outs rather than relying only on type-specific patterns.[9]

Training on Single Strike-out Types This raises the question of whether a single strike-out type may already be sufficient for generalization. To investigate this, we trained YOLO models on ~~HWG~~-written using one strike-out type and evaluated their ability to distinguish the other strike-out types from non-struck-out examples. Each target class was evaluated on a balanced set consisting of all its examples and an equally sized set of randomly sampled non-struck-out examples from ~~HWG~~-collected, since all non-struck-out examples from ~~HWG~~-written had already been used for training. The same sampled non-struck-out examples were used across all target classes to ensure a fair and directly comparable evaluation. Figure 4 presents the resulting F_1 scores as a heatmap. Columns show how well a given target strike-out type is recognized by models trained on different source types, whereas rows show how well a model trained on one type transfers to others. Diagonal entries show performance on held-out examples of the same strike-out type used for training. The results suggest that *wavy* is the strongest source type, whereas *crossed* is the most reliably recognized target type. However, this pattern does not hold in reverse, as *wavy* is also the most difficult target type, meaning that models trained on other single source types transfer least effectively to *wavy*.

Training dataset	Evaluation dataset				
	~~HWG~~-collected	~~HWG~~-SOW	~~HWG~~-written	~~HWG~~-synthetic	SWS
~~HWG~~-collected	(0.97)	.88	.74	.88	.69
~~HWG~~-SOW	.82	(0.94)	.65	.88	.66
~~HWG~~-written	.89	.60	(0.96)	.85	.94
~~HWG~~-synthetic	.38	.49	.13	(1.00)	.26
SWS	.80	.83	.84	.73	(1.00)

Table 8: Macro-F_1 scores for YOLO.

5.3 Performance in a Realistic Application Scenario

In a realistic application scenario, detecting struck-out words alone is not sufficient, since the system must also avoid falsely classifying non-struck-out words as struck-out. Figure 5 illustrates the practical impact of such false positives. In this example, the YOLO model trained on ~~HWG~~-written detects all struck-out instances, but also produces two false positives. If these falsely detected words were removed from the transcript, the sentence would read *... the python are very and invade ...* instead of *... the python are very dangerous and invade ...*. This example motivates evaluating both classes jointly. We therefore report macro-averaged F_1 (see Table 8), which gives equal weight to the struck-out and non-struck-out classes. Consistent with previous cross-dataset results, macro-F_1 scores show

[8] For each type-specific evaluation, we use all samples of the type and the same number of random non-struck-out examples. Exact numerical results are provided in Appendix C.

[9] Detailed precision, recall, and F_1 results for each held-out class are provided in Table 15 in Appendix D.

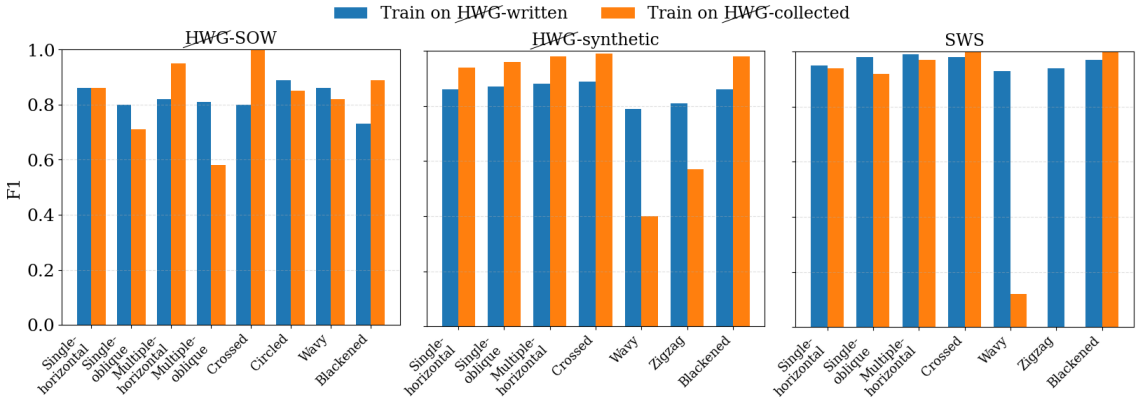


Figure 2: F_1 by strike-out type: training on ~~HWG~~-written vs. ~~HWG~~-collected.

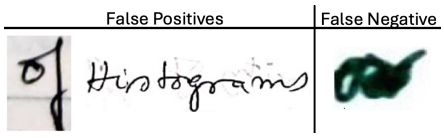


Figure 3: Examples of incorrect predictions for the class *blackened*.

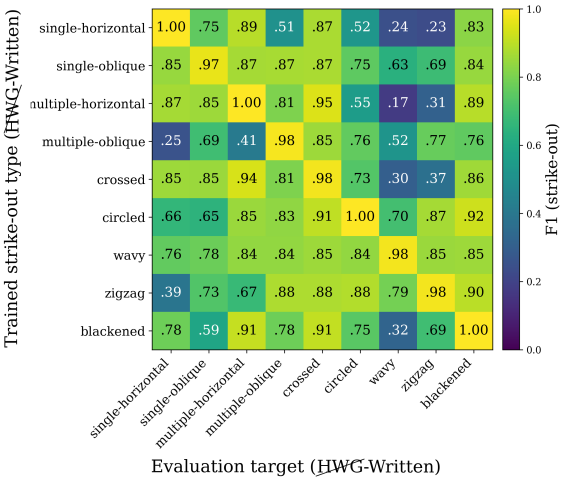


Figure 4: F_1 scores for the struck-out class of YOLO trained on ~~HWG~~-written using a single strike-out type.

that performance strongly depends on the training and target dataset. Averaged over the three real evaluation datasets, SWS (.82) and ~~HWG~~-written (.81) transfer best, while ~~HWG~~-SOW performs worst (.71), suggesting that broader strike-out-type coverage improves robustness.

5.4 Amount of Training Data

As our new datasets provide more samples than previously available public datasets, we can investigate whether additional training data also improves performance in this setting. We train YOLO models on progressively larger subsets of the ~~HWG~~-written

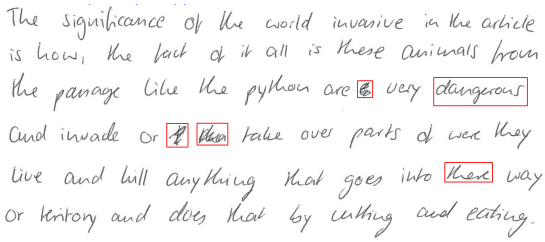


Figure 5: Full-text example in which all words predicted as struck out are highlighted. Note that two are clearly false positives.

dataset, increasing the sample size in steps of 10 and repeating each experiment ten times with different random samples. The resulting models are then evaluated on each individual real dataset. Figure 6 shows the Macro-F_1 learning curves across the real datasets, with shaded areas representing the standard deviation across the ten repetitions for each sample size.[10] Overall, the learning curves show a steep initial improvement and saturate after approximately 50–100 training samples across all datasets.

5.5 Zero-shot VLM Baselines

Finally, we compare our trained classifiers with recent general-purpose vision-language models in a zero-shot setting. We evaluate proprietary models (Gemini 3.1 Flash Lite and Gemini 3.5 Flash) and open-weight models (Mistral Medium 3.5, Qwen3-VL 235B, and Qwen3-VL 32B) on ~~HWG~~-collected. Each model receives a word crop and is prompted to classify it as struck-out or non-struck-out. The results are shown in Table 9.[11]

[10] See Appendix E, Figures 9 and 10, for class-specific learning curves and type-specific F_1 scores.

[11] The recall and precision for the struck-out class are reported in Table 12 in Appendix B.

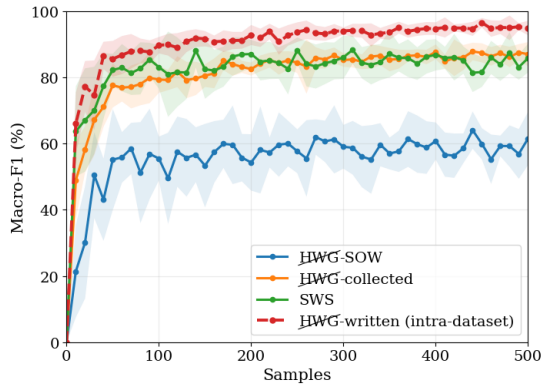


Figure 6: Learning curves for YOLO trained on increasing numbers of HWG-written samples and evaluated on all real datasets ($n = 10$ repetitions). Results for HWG-written are reported on the 15% held-out test set.

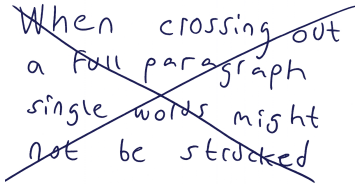


Figure 7: Struck-out paragraph where some words that belong to the deleted content are not directly struck-out.

Gemini 3.5 Flash performs best, achieving a macro-F_1 score of .94 and outperforming the best cross-dataset classifier on HWG-collected. The other VLMs obtain macro-F_1 scores between .78 and .89, which is comparable to classifiers transferred from other real datasets to HWG-collected. Thus, VLMs can be competitive zero-shot alternatives, but their use may be limited by high self-hosting requirements or privacy concerns when using API-based models on sensitive student data. Table 10 reports the per-type F_1 scores on HWG-collected for the best supervised classifier and the best zero-shot VLM.

6 Conclusion

Starting from the observation that publicly available datasets for struck-out handwritten words are scarce, we create several datasets covering multiple strike-out types. Using these datasets, we show in a cross-dataset evaluation with both a CNN-based and a transformer-based classifier that strong intra-dataset performance does not necessarily transfer to robust cross-dataset generalization, suggesting that previous work likely overestimated real-world robustness. We further show that our newly created data improves generalization, especially for rare strike-out types. At the same time, the learning curves indicate early saturation, suggesting that substantially more annotated struck-out words may yield only limited gains.

Model	Training data	$F_1(s)$	Macro-F_1
YOLO	HWG-written	.89	.89
Gemini 3.5 Flash	Zero-shot	.93	.94
Gemini 3.1 Flash Lite	Zero-shot	.88	.89
Mistral Medium 3.5	Zero-shot	.82	.81
Qwen3-VL 235B	Zero-shot	.82	.78
Qwen3-VL 32B	Zero-shot	.87	.88

Table 9: Comparison of the best supervised cross-dataset baseline and zero-shot VLMs on HWG-collected.

Strike-out type	**Samples**	**YOLO**	**Gemini 3.5 Flash**
Single-horizontal	74	0.92	0.95
Single-oblique	105	0.89	0.94
Multiple-horizontal	34	0.88	0.96
Multiple-oblique	2	0.80	1.00
Crossed	69	0.93	0.97
Circled	8	0.94	0.93
Wavy	3	0.86	0.50
Zigzag	10	1.00	0.78
Blackened	97	0.87	0.92

Table 10: F_1 by strike-out type on HWG-collected. YOLO was trained on HWG-written, whereas Gemini 3.5 Flash was evaluated zero-shot.

Limitations

Our study focuses on word-level struck-out detection, enabling cross-dataset experiments but excluding document-level cases such as struck-out lines, longer spans, or paragraphs. In such cases, some words may belong to deleted content without being directly intersected by a strike-out, as shown in Figure 7 (e.g. *out*, *a*, and *single*). We also did not evaluate graph paper, where background grid lines may interfere with strike-out detection. In such cases, line removal may be needed to improve detection instead of discarding the affected word.

Acknowledgments

This work was partially conducted at "CATALPA - Center of Advanced Technology for Assisted Learning and Predictive Analytics" of the FernUniversität in Hagen, Germany. We thank Zhong et al. (2025) for providing the SOW dataset.

References

Chandranath Adak and Bidyut B Chaudhuri. 2014. An approach of strike-through text identification from handwritten documents. In *2014 14th International Conference on Frontiers in Handwriting Recognition*, pages 643–648. IEEE.

Chandranath Adak, Bidyut B Chaudhuri, and Michael Blumenstein. 2017. Impact of struck-out text on writer identification. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 1465–1471. IEEE.

Nilanjana Bhattacharya, Umapada Pal, and Partha Pratim Roy. 2017. Cleaning of online Bangla free-form handwritten text. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 17(1):1–25.

Axel Brink, Harro van der Klauw, and Lambert Schomaker. 2008. Automatic removal of crossed-out handwritten text and the effect on writer verification and identification. In *Document Recognition and Retrieval XV*, volume 6815, page 68150A. International Society for Optics and Photonics.

Bidyut B Chaudhuri and Chandranath Adak. 2017. An approach for detecting and cleaning of struck-out handwritten text. *Pattern Recognition*, 61:282–294.

R. Dheemanth Urs and H. K. Chethan. 2021. A study on identification and cleaning of struck-out words in handwritten documents. In *Data Intelligence and Cognitive Informatics*, pages 87–95, Singapore. Springer Singapore.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*.

Christian Gold, Dario van den Boom, and Torsten Zesch. 2021. Personalizing handwriting recognition systems with limited user-specific samples. In *Document Analysis and Recognition – ICDAR 2021*, pages 413–428, Cham. Springer International Publishing.

Christian Gold, Said Yasin, and Torsten Zesch. 2026. HWG dataset.

Christian Gold and Torsten Zesch. 2020a. Exploring the impact of handwriting recognition on the automated scoring of handwritten student answers. In *2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 252–257.

Christian Gold and Torsten Zesch. 2020b. Handwritten ASAP short answer scoring.

Raphaela Heil, Ekta Vats, and Anders Hast. 2021a. IAM Strikethrough Database.

Raphaela Heil, Ekta Vats, and Anders Hast. 2021b. Strikethrough removal from handwritten words using CycleGANs. In *2021 International Conference on Document Analysis and Recognition (ICDAR)*.

Laurence Likforman-Sulem and Alessandro Vinciarelli. 2008. HMM-based offline recognition of handwritten words crossed out with different kinds of strokes. In *Proceedings of the 11th International Conference on Frontiers in Handwriting Recognition (ICFHR 2008)*.

Urs-Viktor Marti and H. Bunke. 2002. The IAM-database: An English sentence database for offline handwriting recognition. In *International Journal on Document Analysis and Recognition*, volume 5, pages 39–46.

Shivangi Nigam, Adarsh Behera, Manas Gogoi, Shekhar Verma, and P. Nagabhushan. 2023. Strike off removal in Indic scripts with transfer learning. *Neural Computing and Applications*, 35:1–17.

Shivangi Nigam, Adarsh Prasad Behera, Shekhar Verma, and P. Nagabhushan. 2024. Deformity removal from handwritten text documents using variable cycle GAN. *International Journal on Document Analysis and Recognition (IJDAR)*, 27(4):615–627.

Hiqmat Nisa. 2023. Handwritten Synthetic Dataset from the IAM.

Hiqmat Nisa, James A. Thom, Vic Ciesielski, and Ruwan Tennakoon. 2019. A deep learning approach to handwritten text recognition in the presence of struck-out text. In *2019 International Conference on Image and Vision Computing New Zealand (IVCNZ)*, pages 1–6.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, and 7 others. 2024. DINOv2: Learning robust visual features without supervision. *Preprint*, arXiv:2304.07193.

Arnab Poddar, Akash Chakraborty, Jayanta Mukhopadhyay, and Prabir Kumar Biswas. 2021. Detection and localisation of struck-out-strokes in handwritten manuscripts. In *Document Analysis and Recognition – ICDAR 2021 Workshops*, pages 98–112, Cham. Springer International Publishing.

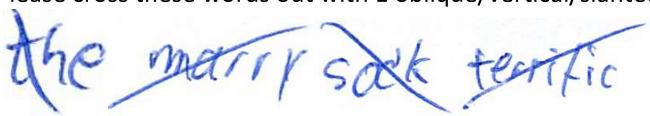
Ranjan Sapkota, Rahul Harsha Cheppally, Ajay Sharda, and Manoj Karkee. 2026. YOLO26: Key architectural enhancements and performance benchmarking for real-time object detection. *Preprint*, arXiv:2509.25164.

Palaiahnakote Shivakumara, Tanmay Jain, Umapada Pal, Nitish Surana, Apostolos Antonacopoulos, and Tong Lu. 2022. Text line segmentation from struck-out handwritten document images. *Expert Systems with Applications*, 210:118266.

Palaiahnakote Shivakumara, Tanmay Jain, Nitish Surana, Umapada Pal, Tong Lu, Michael Blumenstein, and Sukalpa Chanda. 2021. A connected component-based deep learning model for multi-type struck-out component classification. In *Document Analysis and Recognition – ICDAR 2021 Workshops*, pages 158–173, Cham. Springer International Publishing.

Dajian Zhong, Shivakumara Palaiahnakote, Umapada Pal, and Yue Lu. 2025. Struck-out handwritten word detection and restoration for automatic descriptive answer evaluation. *Signal Processing: Image Communication*, 130:117214.

A Participant Instructions

Type	Description and example
None	No further action required
Single-Horizontal	Please cross these words out with 1 horizontal line 
Single-Oblique	Please cross these words out with 1 oblique/vertical/slanted line 
Multiple-Horizontal	Please cross these words out with several (minimum of 2) horizontal lines 
Multiple-Oblique	Please cross these words out with several (minimum of 2) oblique/vertical/slanted lines 
Crossed	Please cross these words out 
Circled	Please circle this word out 

Wavy	Please use waves to cross these words out 
Zigzag	Please use vertical zigzags to cross these words out 
Blackened	Please blacken these words 

Figure 8: Instruction sheet used for data collection.

B Precision, Recall, and F_1 for Struck-Out Word Detection

Training dataset	Model	Evaluation dataset														
		~~HWG~~-collected			~~HWG~~-SOW			~~HWG~~-written			~~HWG~~-synthetic			SWS		
~~HWG~~-collected	YOLO	0.98	0.96	0.97	0.79	0.76	0.77	0.99	0.84	0.91	0.97	0.79	0.87	1.00	0.68	0.81
	DINO	0.95	0.93	0.94	0.63	0.78	0.69	1.00	0.79	0.88	0.96	0.78	0.86	1.00	0.56	0.72
~~HWG~~-SOW	YOLO	0.97	0.66	0.79	0.84	0.90	0.89	1.00	0.74	0.85	0.99	0.76	0.86	1.00	0.64	0.78
	DINO	0.96	0.78	0.86	0.77	0.81	0.83	1.00	0.67	0.80	0.96	0.60	0.74	1.00	0.53	0.70
~~HWG~~-written	YOLO	0.84	0.95	0.89	0.20	0.95	0.33	0.99	0.99	0.99	0.85	0.86	0.85	0.99	0.96	0.98
	DINO	0.76	0.99	0.86	0.31	0.83	0.45	0.99	0.98	0.98	0.79	0.94	0.86	0.99	0.92	0.95
~~HWG~~-synthetic	YOLO	0.72	0.04	0.08	0.00	0.00	0.00	1.00	0.04	0.07	1.00	1.00	1.00	1.00	0.12	0.21
	DINO	0.94	0.67	0.78	0.39	0.73	0.51	1.00	0.79	0.89	1.00	1.00	1.00	1.00	0.63	0.77
SWS	YOLO	0.72	0.99	0.83	0.57	0.85	0.68	0.98	0.95	0.96	0.67	0.99	0.80	1.00	1.00	1.00
	DINO	0.83	0.97	0.90	0.28	0.90	0.43	0.99	0.93	0.96	0.81	0.82	0.82	1.00	0.96	0.98

Table 11: Precision, recall, and F_1 scores for the struck-out class across all training and evaluation datasets.

Model	F_1	R	P
Gemini 3.5 Flash	.93	.93	.94
Gemini 3.1 Flash Lite	.88	.79	.99
Mistral Medium 3.5	.82	.90	.76
Qwen3-VL 235B	.82	.97	.71
Qwen3-VL 32B	.87	.78	.98

Table 12: Zero-shot VLM performance on ~~HWG~~-collected for the struck-out class.

C Strike-out Type Analysis

Strike-out type	~~HWG~~-collected	~~HWG~~-SOW	~~HWG~~-synthetic	SWS
Single-horizontal	0.92	0.86	0.86	0.95
Single-oblique	0.89	0.80	0.87	0.98
Multiple-horizontal	0.88	0.82	0.88	0.99
Multiple-oblique	0.80	0.81	–	–
Crossed	0.93	0.80	0.89	0.98
Circled	0.94	0.89	–	–
Wavy	0.86	0.86	0.79	0.93
Zigzag	1.00	–	0.81	0.94
Blackened	0.87	0.73	0.86	0.97

Table 13: F_1 by strike-out type for models trained on ~~HWG~~-written and evaluated on the other datasets.

Strike-out type	~~HWG~~-written	~~HWG~~-SOW	~~HWG~~-synthetic	SWS
Single-horizontal	0.97	0.86	0.94	0.94
Single-oblique	0.89	0.71	0.96	0.92
Multiple-horizontal	0.98	0.95	0.98	0.97
Multiple-oblique	0.89	0.58	–	–
Crossed	0.97	1.00	0.99	1.00
Circled	0.90	0.85	–	–
Wavy	0.54	0.82	0.40	0.12
Zigzag	0.74	–	0.57	0.00
Blackened	0.97	0.89	0.98	1.00

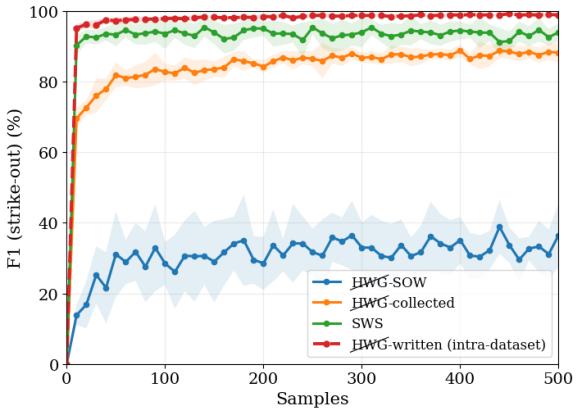
Table 14: F_1 by strike-out type for models trained on ~~HWG~~-collected and evaluated on the other datasets.

D Held-out Strike-out Type Analysis

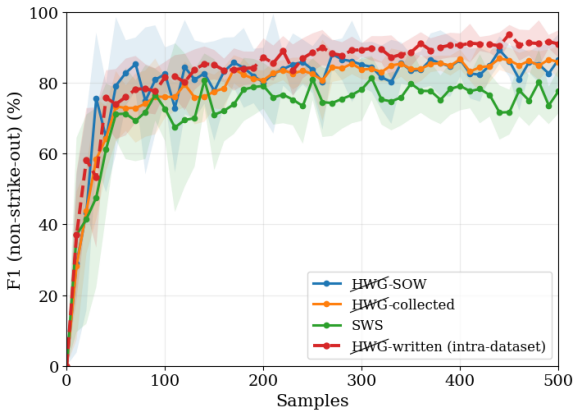
Held-out class	Precision	Recall	F1
single-horizontal	0.83	0.98	0.90
single-oblique	0.83	0.95	0.89
multiple-horizontal	0.82	1.00	0.90
multiple-oblique	0.81	0.99	0.89
crossed	0.83	1.00	0.91
circled	0.79	1.00	0.88
wavy	0.84	0.91	0.88
zigzag	0.79	1.00	0.88
blackened	0.79	1.00	0.88

Table 15: Results for held-out struck-out classes.

E Training Dynamics



(a) F_1 for the struck-out class



(b) F_1 for the non-struck-out class

Figure 9: Class-specific learning curves corresponding to Figure 6.

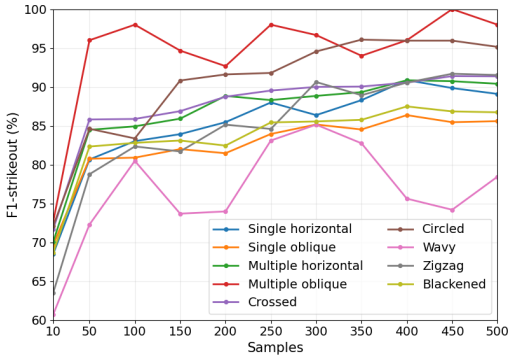


Figure 10: F_1 scores for the struck-out class for YOLO models trained on ~~HWG~~-written data and evaluated on different types of ~~HWG~~-collected data.