ACADEMIA International Journal for Social Sciences
Volume 5, Issue 3(s11), 2026 ISSN-L (Online): 3006-6638

Word Segmentation in Urdu Text from News Video Tickers Using Deep Learning

Abdul Moiez
moiez536@gmail.com
Dept of CS&IT MUST, Mirpur AJK, Pakistan

Hifsa Ali
qureshihifsa92@gmail.com
Dept of CS&IT MUST, Mirpur AJK, Pakistan

Sobia Sher
sobia.csit@gmail.com
Dept of CS&IT MUST, Mirpur AJK, Pakistan

Corresponding Author: Abdul Moiez moiez536@gmail.com

Received: 14-02-2026 **Revised:** 06-03-2026 **Accepted**: 14-03-2026 **Published:** 26-03-2026

Abstract

Automated word segmentation in Urdu text is a key problem in Optical Character Recognition (OCR) and Natural Language Processing (NLP) workflows because of its ligature-based Nastaliq script, in which Urdu is written from right to left. In broadcast news media, Urdu text emerges as a scrolling ticker integrated into video frames. Proper word segmentation is important for Urdu text recognition in video tickers. This paper introduces a deep learning-based method for segmentation of Urdu text tokens from news video tickers. We employed Fast Region-based Convolutional Neural Network (Fast R-CNN) with three transfer-learned feature extractors, a standard CNN, AlexNet, and VGG16, and train six detector configurations on the subsets of a custom annotated dataset, the Custom Urdu dataset (CUDS). CUDS comprises 2,515 scene images containing 27,199 labeled Urdu word instances collected from four Pakistani news television channels. Experimental results show that the best-performing model, F_R-CNN (CNN backbone, CUDS), achieves a precision of 1.0 and recall of 0.98 at 30 training epochs, while F_R-CNN (AlexNet, CUDS2) achieves precision and recall of 1.0. Comparative analysis across all six models demonstrates the strong suitability of Fast R-CNN for Urdu word segmentation in broadcast ticker footage.

Keywords:*Urdu word segmentation, news video tickers, Fast R-CNN, Nastaliq script, deep learning, transfer learning, broadcast OCR.*

INTRODUCTION

Urdu is the national language of Pakistan, and more than170 million people worldwide speak Urdu.It is written in a Nastaliq form of the Perso-Arabic script, distinguished by its flowing and calligraphic style in which characters are connected in a right-to-left manner. In contrast to Latin script languages, where word boundaries are clearly separated by white-space, Urdu text does not utilize spaces uniformly to separate words. Instead, words consist of one or more ligatures, a context-dependent joining of character groups, making the automatic recognition of word boundaries a nontrivial challenge Urdu text is continuously transmitted by Pakistani news television channels in the form of scrolling tickers, bottom-aligned horizontal overlays carrying breaking news, weather updates, stock prices, and sports scores .Automatic extraction and interpretation of this text have importantapplications in media monitoring, news archiving, accessibility features for users with hearing impairments,and retrieving real-time information . Word segmentation is the primary phase of this framework; without reliable recognition of individual words in the ticker area, subsequent text recognition is adversely affected.Although it is of significant importance in Urdu text processing , related work on video text segmentation has primarily focused on Latin and Chinese and to a lesser Extent, Arabic scripts . Urdu specific word segmentation from videos has received limited attention in

the literature ,and no prior work has applied FAST R-CNN with structured backbone comparison to this task.This paper addresses this gap by presenting a targeted study of word-level segmentation from Urdu news ticker footage using a deep learning.

The key contributions of this paper are: (1) a manually annotated dataset of real Urdu news ticker frames with word-level bounding box labels; (2) six trained word detection models using Fast R-CNN with CNN, AlexNet, and VGG16 backbones across two data configurations; and (3) a systematic comparative evaluation highlighting the effect of backbone choice, training duration, and dataset size on word segmentation accuracy.

RELATED WORK

Textdetectioninvideo

We categorized Text detection in video into three approaches: region-based, connected component-based, and texture-based approaches. Early methodologies mainly focus statistical image Properties.For instance, Ma et al. [1] developed a localization method using a Radial Basis Function Neural Network (RBFNN) to locate text . Gómez et al. [2] employed Maximally Stable Extremal Regions (MSER) for real-time text tracking. Later, Tian et al. [3]proposed Text Flow, to combine connected components with cascade boosting to detect text in natural scenes.With the advancement of deep learning, researchers start using more advanced techniques. ; Zayene et al. [4] use Stroke Width Transform (SWT) and Convolutional Auto-Encoder(CAE) to detect Arabic text in videos, while Jain et al. [5] applied a hybrid CNN-RNN architecture for Arabic text recognition in unconstrained videos. Several methods are developed to mainly focus on Latin, Chinese or standard Arabic scripts than other scripts.

Arabic-script and Urdu text detection

Urdu text detection is challenging because of the complex Nastaliq script, where letters are connected and their shapes can change depending on the context. Jamil et al. 66 introduced an edge-based method to locate artificial Urdu text in video frames. Later, Unar et al. 77 combined Sobel and Canny edge detection with MSER and an SVM classifier to detect artificial Urdu text. Mirza et al. 88 also explored Urdu text detection by extracting textural features using Garbo filters and Curve-let transforms. Although these traditional methods have contributed to Urdu text detection, there is still limited research on deep learning-based Urdu word and ligature segmentation in broadcast videos. In particular, the use of region-proposal networks, such as Fast R-CNN 99, for this task has received little attention.

DATASET

CustomUrduDataset(CUDS)

Most publicly available Urdu text datasets contain document images and do not work well for the video ticker domain.To solve this, we developed the Custom Urdu Dataset (CUDS) specifically for Urdu news ticker word recognition. We gathered video footage from four Pakistani news television channels broadcasting in Urdu. From this video footage, we extracted frames where news tickers were visible and manually annotated with word-level bounding boxes.

We annotated the video using ground truth labler app available in MATLAB. this app provides a '.mat' file consisting of images and ground truth values.

1. CUDS1 (Full Corpus): Utilizes the entire dataset of 2,515 images (split 60% for training and 40% for testing) to evaluate detection across high variations in Urdu font style, background color, and scrolling speed.

2. CUDS2 (Held-Out Split): A targeted subset of 1,270 single-broadcast frames. This is divided into 850 training frames and 420 completely held-out test frames to evaluate how the models generalize tounseen footage.

Fig 1. Sentence Level Sample Images of Cuds

Table 1. Cuds Dataset Statistics

Subset	Train images	Test images	Word instances
CUDS	10880	16319	27,199

PROPOSED METHODOLOGY

Overview

Our word segmentation framework processes cropped ticker region images and generates word-level bounding boxes. The detector used is Fast R-CNN, which is fast to train and test as compared to RCNN and SPPnet. We traied and tested three different models for detection using CNN, Alexnet and VGG-16.The framework consists of three main components: preprocessing, region proposal generation and classification and regression.

Architectural Selection Rationale

Single-stage detectors such as YOLO and region-based methods like Faster R-CNN are commonly used for text detection.However, segmenting Urdu Nastaliq text is challenging because of its unique writing style. Unlike Latin scripts, Urdu is written in cursive style often overlapping to make text more complex.

Gridbased model often find it difficult to locate word boundries, especially when letters from neighbouring words overlap. Fast R-CNN was selected because it uses selective search to find possible text regions based on visual features like color, edges and texture before classification. This approach helps us to understand how traditional region proposal methods detect the word boundaries of Nastaliq text having no clear space.

Pre-processing

Input frames are resized to 224x224 pixels, consistent with the input requirements of the AlexNet and VGG16 backbones. Color normalization is applied channel-wise using the ImageNet mean and standard deviation.

Fast R-CNN Architecture

Fast R-CNN processes an input image by passing

it through a convolutional backbone to produce a shared feature map. Selective search generates approximately 2,000 region proposals per image, which are projected onto the feature map using RoI (Region of Interest) pooling. Each pooled region is passed through fully connected layers to produce two outputs: a softmax classification score (word vs. background) and a bounding box regression offset.

We experiment with three backbone feature extractors. The baseline CNN is a four-layer convolutional network trained from a random initialization. AlexNet [10] provides a five-layer convolutional feature hierarchy pre-trained on ImageNet. VGG16 provides a deeper thirteen-layer convolutional hierarchy, also pre-trained on ImageNet [11]. We

applied transfer learning for both AlexNet and VGG16 by fine-tuning all layers on the Urdu word detection task from the Image Net-pretrained weights.

Model configurations

Three word-detection models are trained as follows: F_WDM1(CNN backbone, CUDS), F_WDM2(AlexNet, CUDS1), F_WDM3(VGG16, CUDS1),. Each model is evaluated at different stages to understand how the model learns. CNN/AlexNet models are evaluated at 30, 60, and 90 epochs, and VGG16 models are evaluated at 15, 25, 30, 50, and 90 epochs to capture full learning dynamics.

Implementation and Training Details:

All models were implemented and trained using MATLAB platform on workstation equipped with an Intel Core i5 processor, 20GB RAM, and a NVIDIA GTX 1050 GPU (2GB). The networks were optimized using Stochastic Gradient Descent with Momentum (SGDM). Training was executed across variable checkpoints up to 90 epochs to capture full learning dynamics and carefully monitor boundary over-segmentation.

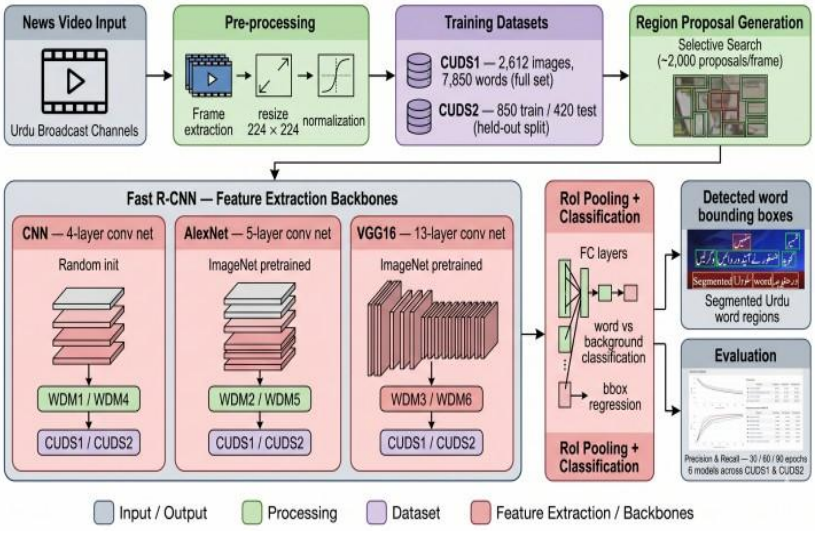

Figure 2: architectural diagram tailored for word segmentation

EXPERIMENTAL RESULTS

Evaluationmetrics

Precision (P) and Recall (R) are used to measure the performance, they are defined as:

P = (Total number of correctly detected word regions) / (Total number of detected word regions)

R = (Total number of correctly detected word regions) / (Total number of ground truth word regions)

A detected region is considered correct if its intersaction over union (IoU) with the closest ground truth box surpasses 0.5.

Resultson CUDS

F_WDM1 (CNN), F_WDM2 (AlexNet), and F_WDM3 (VGG16) models were trained and tested on CUDS. Precision and recall across training epochs are shown in Table 2.

Table 2: Word detection results on CUDS

Madel	Backbone	Epoch	Precision	Recall	F1 Score
F_WDM 1	CNN	30	0.90	0.91	0.90
F_WDM 1	CNN	60	0.95	0.94	0.95
F_WDM 1	CNN	90	1.00	0.98	0.99
F_WDM 2	AlexNet	30	0.92	0.90	0.91
F_WDM 2	AlexNet	60	0.95	0.90	0.92
F_WDM 2	AlexNet	90	1.00	0.90	0.95
F_WDM 3	VGG16	15	0.88	0.96	0.92
F_WDM 3	VGG16	30	0.89	0.99	0.94
F_WDM 3	VGG16	50	0.90	1.00	0.95
F_WDM 3	VGG16	90	0.88	0.96	0.92

F_WDM1 achieves a precision of 1.0 and a recall of 0.98 at 90 epochs showing the best balance between a precision and recall, making it suitable model for CUDS. F_WDM2 also achieves the same level of precision by 90 epochs;but recall is only 0.90. This means that even the model detect the word accurately,but it is difficult for modeln to detect complex and unclear word boundaries. In contrast, F_WDM3 (VGG16) achieves a recall of 1.0 at 50 epochs, but its precision stops at 0.90 indicating that the deeper model produce more false word detections.

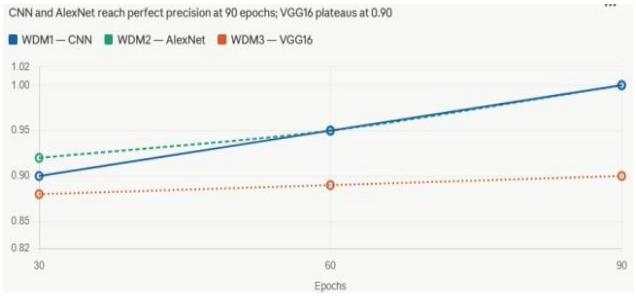

Figure 3: Precision learning curves for CUDS1 models (WDM1–WDM3): CNN and AlexNet both reach 1.0, while VGG16 plateaus at 0.90.

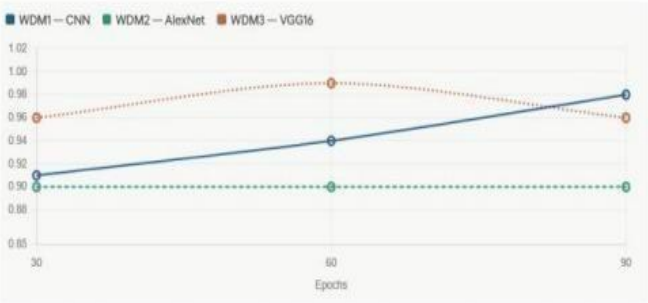

Figure 4: Recall learning curves on CUDS: WDM1 (CNN) start at 0.98 and at 90 epochs.

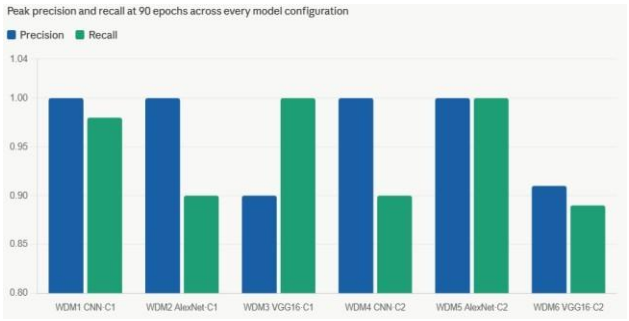

Figure 5: A side-by-side comparison of the best epochs of all six models makes it clear right away that WDM5 is the best and WDM6 (VGG16, CUDS2) is the worst.

DISCUSSION

While analyzing the results, we see that precision and recall improve by increasing training epochs for CNN and AlexNet backbones. However, recall reaches its maximum at 60 epochs and then decreases at 90 epochs on CUDS1. This behavior indicates the VGG16 models reflect overfitting.

Analysis of misclassifications results shows that the lower precision of VGG16 may be related to the structure of Urdu text.Due to its deeper architecture, VGG16 model tends to divide.The model oftenly detect individual part of word instead of identifying the whole word with single rectangular box. In dense news tickers, the ending part of one word may extend beneath the succeeding word. CNN and AlexNet handle these overlapping regions using non-maximum suppression (IoU > 0.5), which helps to remove duplicate detections. In contrast, VGG-16 has difficultly handling these overlaps, which result in redundant word segmentation.This result suggests that shallower feature hierarchies provide a balance between spatial resolution and semantic abstraction for tightly spaced Nastaliq tickers.

For practical deployment, we suggest F_WDM1 when completely annotated dataset is available, and F_WDM2 when a held-out test split is required.

The reported high performance metrics (P=1.0, R=0.98 and P=1.0, R=1.0) should be interpreted in light of the evaluation protocol. For CUDS, the models were trained and tested on the split data sets, which provides a more realistic estimates of generalization, and the perfect score of F_WDM1 on its test split shows the strong robustness to unseen ticker frames within the same broadcast domain.

CONCLUSION

The paper introduced a specialized deep learning approach to word segmentation of Urdu in news video tickers. We employ Fast R-CNN with CNN, AlexNet and VGG16 to train and evaluate three detector configurations on a custom annotated dataset based on real Pakistani news broadcast footage.

F_WDM5 (Alexnet, CUDS) was the best performing model because it got the perfect precision and recall on a test split that was kept separate.

The result indicates that the FAST R-CNN that uses AlexNet features is the most stable and reliable model for Urdu text ticker. Even with less training data, it was better than both shallower CNNs and the deeper VGG16 architectures. This proposed work provide a foundation for future research of Urdu word segmentation in broadcast environments.

Two main improvements are suggested for future work : (1) a sequence-to-sequence OCR module can be added to convert word images into Unicode Urdu text (2) to extend the CUDS dataset by adding challenging video samples from various news channels, nighttime broadcasts, and low-quality streams. This will help evaluate the model under different conditions and test its performance in real-world situations.

REFERENCES

X. Ma, W. Y. Ng, P. K. Chan, and D. S. Yeung, "Video text detection and localization based on localized generalization error model," *IEEE TransactionsonSystems,Man,andCybernetics*, 2010.

L. Gómez and D. Karatzas, "MSER-based real-time text detection and tracking," *International Conference on Pattern Recognition (ICPR)*, 2014.

S. Tian, Y. Pan, C. Huang, S. Lu, K. Yu, and C. L. Tan, "Text Flow: A Unified Text Detection System in Natural Scene Images," *IEEE International Conference on ComputerVision(ICCV)*, 2015.

O. Zayene, M. Seuret, S. M. Touj, J. Hennebert, and N. E. Ben Amara, "Text detection in Arabic news videos based SWT operators and convolutional auto-encoders," *IEEE International Conference on DocumentAnalysisandRecognition(ICDAR)*, 2016.

M. Jain, M. Mathew, and C. V. Jawahar, "Unconstrained Scene Text and Video Text Recognition for Arabic Script,"*Document Analysis Systems (DAS)*, 2016.

A. Jamil, A. Siddiqi, I. Arif, and F. Raza, "Edge-based Features for Localization of Artificial Urdu Text in Video Images," *International Conference on Document Analysis and Recognition*, 2011.

S. Unar, A. H. Jalbani, M. M. Jawaid, M. Shaikh, and A. A. Chandio, "Artificial Urdu Text Detection and Localization from Individual Video Frames," *Mehran University Research Journal of Engineering & Technology*, 2018.

A. Mirza, M. Fayyaz, Z. Seher, and I. Siddiqi, "Urdu Caption Text Detection using Textural Features," *International Conference on Frontiers of InformationTechnology(FIT)*, 2018.

R. Girshick, "Fast R-CNN," *EEEInternationalConferenceonComputer Vision(ICCV)*, pp. 1440-1448, 2015.

A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84-90, 2017.

K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition" arXiv preprint arXiv:1409.1556, 2014.