DeepHCoref: A Deep Neural Coreference Resolution for Hindi Text

Kusum Lata[1], **Pardeep Singh**[2], **Kamlesh Dutta**[2], **Abhishek Kanwar**[2]
[1]Sharda University, Greater Noida, India
ranapoo@gmail.com, kusumlata.10@sharda.ac.in
[2]Department of Computer Science Engineering,
National Institute of Technology, Hamirpur, Himachal Pradesh, India
{pardeep,kd,20dcs013}@nith.ac.in

Abstract

Coreference Resolution is the process of detecting a cluster of mentions that point to the same entity. This paper presents the Coreference Resolution system for Hindi based on Bi-GRU-CNN and Biaffine classifier with IndicBERT and MuRIL BERT. The motivation behind this work is the scarcity of resources available for Hindi and to diminish the dependency on the external parser and hand-crafted feature used by the previous Coreference resolution model in the Hindi language. The coreference annotated dataset is used for the Hindi language, containing 3.6K verbalizations and 78K tokens from the news article domain. The experimental results received are promising in the form of Precision, Recall, and F-measure.

1 Introduction

Coreference Resolution (CR) is the task of creating a link between the referring expression and the referent entity. The Coreference Resolution will enhance the performance of numerous Natural Language Processing (NLP) applications viz. Machine Translation, Question Answering, Chatbots, Text Summarization, etc. The existing Coreference Resolution system (Haghighi and Klein, 2009; Lee et al., 2011; Björkelund and Kuhn, 2014; Durrett and Klein, 2013; Aloraini et al., 2020) divided the Coreference Resolution process into two steps: Mention detection that find out all the mentions such as named entities, pronominal, and nominal entities available in the text, and second step, creating a cluster of mentions that point to same real-world entities. We explain the concept of the CR with the help of the following example SH1:

SH1[1]: फिल्म महोत्सव में प्रकाश झा की नई फिल्म अपहरण का भी प्रीमियर होना है। गंगाजल के बाद उसकी यह किसी अलग विषय पर बनी दूसरी फिल्म है।

SE1[2]: *Prakash Jha's new film Apaharan is also to premiere at the film festival. This is his second film on a different subject after Gangajal.*

SHI1[3]: *Prakash Jha kee naee film apaharan ka bhee film mahotsav mein preemiyar hona hai. Gangaajal ke baad usakee yah kisee alag vishay par banee doosaree film hai.*

In this example of the sentence (in Hindi), **SH1**, the available mentions in this sentence after applying the mention detection step are:

फिल्म महोत्सव *(film festival /film Mahotsav)*, प्रकाश झा *(Prakash Jha)*, नई फिल्म *(naee film /new film)*, अपहरण *(apaharan)*, उसकी *(his /usakee)*, यह *(this /yah)*, दूसरी फिल्म *(second film /doosaree film)*, गंगाजल *(Gangaajal)* .

The mentions फिल्म महोत्सव *(film Mahotsav)*, नई फिल्म *(naee film)*, and दूसरी फिल्म *(doosaree film)* are nominal mentions. The mentions प्रकाश झा *(Prakash Jha)*, अपहरण *(apaharan)*, and गंगाजल *(Gangaajal)* are named mentions. The mentions उसकी *(usakee)* and यह *(yah)* are pronominal mentions.

The step of coreference resolution process for sentence SH1, is shown in Figure 1a and 1b. प्रकाश झा *(Prakash Jha)*, उसकी *(his /usakee)* are in one cluster. And similarly, नई फिल्म *(naee film)*,यह *(yah)*, and दूसरी फिल्म *(doosaree film)* are in the same cluster.

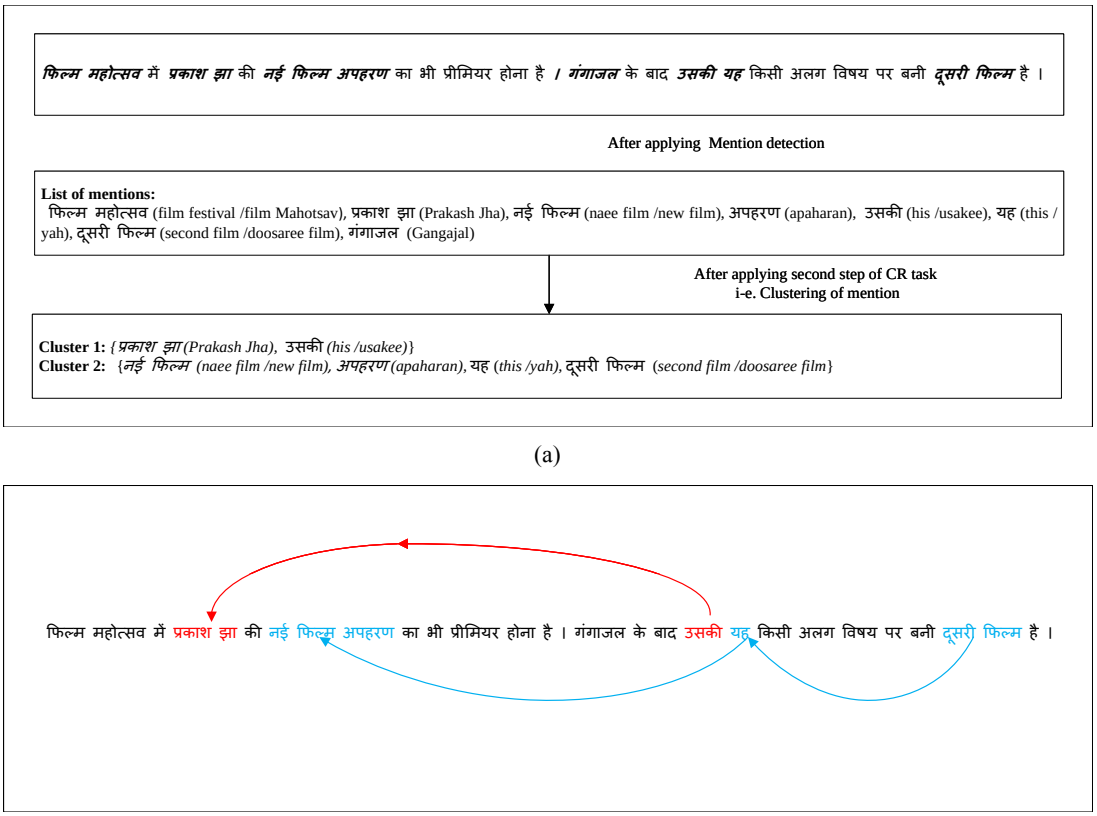
There are many shared task datasets such as ONTONOTES, CoNLL-2011/2012 exist for the English language prominently as discussed by authors Sukthanker et al. (2020); Stylianou and Vlahavas (2021); Lata et al. (2021). In addition, the CRAC shared tasks Žabokrtskỳ et al. (2022, 2023) have made substantial contributions to recent work in multilingual Coreference Resolution. The CRAC 2023 shared task for several languages, including Catalan, Czech, English, French, Ger-

[1]SH: Sentence in Hindi

[2]SE: Sentence in English

[3]SHI: Sentence in Hinglish (Roman Gloss for Hindi)

Proceedings of the Seventh Workshop on Computational Models of Reference, Anaphora and Coreference (CRAC 2024), pages 67–77



(a)

(b)

Figure 1: Coreference Resolution Process for sentence SH1

man, Hungarian, Lithuanian, Norwegian, Polish, Russian, Spanish, and Turkish, is available. There exists significant work on the deep learning-based Coreference Resolution model that has recently shown state-of-the-art performance for the English language. On the other hand, hardly little study has been done on the Coreference Resolution system for the Hindi Language such as Vasantlal (2017); Mishra et al. (2024)

Challenges in Hindi Language: One of the main reasons behind the lack of research in this area could be that numerous hurdles exist in Hindi language viz. no capitalization, free word order, lack of labeled data, being morphologically rich, ambiguity in proper nouns, and insufficiency of linguistic resources which need to be acknowledged while developing Coreference Resolution model.

1. Because Hindi has a flexible word order, it is possible to change the Subject-Object-Verb (SOV) structure without affecting the meaning. Due to this variety, it may be challenging for neural models to develop consistent patterns for coreference resolution because of the wide variations in how entities are positioned in relation to pronouns. When attempting to resolve coreferences in Hindi, neural models must be more flexible than English, where they can frequently rely on more rigid syntactic patterns. In place of spatial clues, this calls for a greater dependence on context-based learning. Attention-based models such as transformers are more appropriate for this task, however they still have issues with word order diversity.

2. As a pro-drop language, Hindi allows subject pronouns to be removed when circumstances suggest they should. It can be challenging for neural models to infer dropped pronouns from the surrounding context in the absence of explicit markers. Implicit references that aren't explicitly stated in the text must be understood by the model. Since neural networks usually rely on explicit tokens for prediction, they may find it difficult to resolve references effectively in sentences when subjects or objects are absent. In order to capture latent references, models must possess a high contextual awareness, which necessitates the integration of mechanisms such as attention. Hindi language displays intricate morphological variations according to case, gender, and number. This results in a vast range of surface forms for verbs, pronouns, and nouns. Given the diversity of forms, it might be dif-

ficult for neural models to learn to link several morphological variations of the same coreferent entity.

Hindi pronouns like वह *(vaha)*, which might signify "he","she," "it," or "that," are sometimes unclear. Depending on the context, a pronoun can be used to refer to several genders, numbers, or even inanimate objects. It is necessary for neural models to precisely distinguish between these allusions based on context, which is frequently more intricate in Hindi. For example:

SH2: लालू की पत्नी पूर्व मुख्यमंत्री राबड़ी देवी के सबसे छोटे भाई सुभाष ने राजद के वरिष्ठ नेता और पूर्व मंत्री जगदानंद सिंह पर आरोप लगाया कि वह पार्टी हितों के खिलाफ काम कर रहे हैं।

SHI2:*laaloo kee patnee poorv mukhyamantree raabadee devee ke sabase chhote bhaee subhaash ne raajad ke varishth neta aur poorv mantree jagadaanand sinh par aarop lagaaya ki vah paartee hiton ke khilaaph kaam kar rahe hain.*

SE2: *Subhash, the youngest brother of Lalu's wife and former chief minister Rabri Devi, accused senior RJD leader and former minister Jagadanand Singh that he is working against the interests of the party.*

In this example, वह *(vaha)*, refers to वरिष्ठ नेता *(varishth neta)*, पूर्व मंत्री जगदानंद सिंह *(poorv mantree jagadaanand singh)*, which is masculine. The pronoun वह *(vaha)* needs to match the gender of its antecedent. Even if the antecedent पूर्व मुख्यमंत्री राबड़ी देवी *(poorv mukhyamantree raabadee devee)* is feminine, the pronoun would still be वह *(vaha)*, however the context would specify the right referent. The gender agreement makes it more difficult to resolve coreferences because the algorithm has to accurately identify the antecedent's gender.

3. The other reason could be the restricted availability of training data in the appropriate format which is required for the specific task.

Contribution of the paper: The key contributions of the paper are as follows: We propose a neural network-based Coreference Resolution system to create clustering of mentions in Hindi text by utilizing Bi-GRU along with transformer-based IndicBERT and MuRIL BERT model and character-level embedding.

We compare the performance of Coreference Resolution system by employing language model with mBERT.

In this paper, our model aims to diminish the need for hand-crafted features and external dependency parsers. We compare the performance of Rule-based Coreference Resolution, a neural-based state-of-the-art Coreference Resolution model for the Hindi language with our model.

The rest of the paper is organized into the following sections. Section 2 contains a comprehensive background of models for Coreference Resolution that have been created or the Related Work done in the area. Section 3 describes the Proposed Approach for the work. Section 4 will expound on the Experimental Evaluation, and Section 5 verbalizes the Conclusion and Future Scope of our work.

2 Related Work

The Coreference Resolution task has been exhaustively researched in literature prominently for the English language. Firstly, we discuss the work related to Coreference Resolution for the English language followed by work for the Hindi Language.

2.1 Coreference Resolution for English

Recently, many researchers (Sukthanker et al., 2020; Lata et al., 2021; Stylianou and Vlahavas, 2021) have conducted in-depth surveys for Coreference Resolution. Various approaches are utilized for Coreference Resolution tasks, and Sukthanker et al. (2020) classified these approaches into three categories: Rule-based, Statistical and machine learning-based, and Deep learning-based. The author also analyzed resolution algorithms on different datasets. Stylianou and Vlahavas (2021) reviewed the most recent neural Coreference Resolution approaches, specifically those involving deep learning techniques. The neural Coreference Resolution approach was prominently employed and analyzed in the English language by different authorsWiseman et al. (2015); Clark and Manning (2016b,a); Lee et al. (2017, 2018). The coreference resolution task can be performed in a pipeline manner (Clark and Manning, 2016a) or a joint manner (Daumé III and Marcu, 2009).

Lee et al. (2017) proposed an end-to-end neural Coreference Resolution system that achieved state-of-the-art performance by combining two tasks: mention detection and Coreference Resolution. Their system automatically learned features for detecting mentions using Bi-directional LSTM and did not rely on hand-crafted features. They employed Glove embeddings and character embeddings to represent words and evaluated their system's performance on the CoNLL-2012 shared

task for English coreference resolution, reporting F1-measures of 77.20% (MUC), 66.60% (B3), 62.60% (CEAF), and an overall F1 of 68.80%.

Building on this, Lee et al. (2018) extended their work by using ELMO embeddings Peters et al. (2018) and second-order inference, improving performance by 0.4 percentage points. Kantor and Globerson (2019) further modified Lee et al.'s model to provide entity-level representation by summing mention representations within a cluster and employed BERT embeddings (Devlin, 2018) instead of ELMO. Joshi et al. (2019) introduced BERT-large, improving the model's performance, while Joshi et al. (2020) later introduced SpanBERT to better represent and predict text spans, resulting in a 2.7% improvement over their previous model. Wu et al. (2020) developed CorefQA with SpanBERT, recasting coreference resolution as a query-based span prediction problem in question answering. They pre-trained the model using question-answering corpora and evaluated it on the CoNLL English shared task dataset, surpassing previous state-of-the-art models(Joshi et al., 2019, 2020) by 0.3% and 3.5%, respectively.

2.2 Coreference Resolution for Hindi

Several researchers have adapted approaches for pronominal resolution in Hindi text from the methods used in English. Prasad and Strube (2000) implemented the centering theory for resolving pronominal references in Hindi, while Dutta et al. (2008) adapted Hobbs Algorithm Hobbs (1978) to handle Hindi's free word order and grammatical nuances. Uppalapu and Sharma (2009) extended the centering theory-based algorithm by managing entities in present and prior utterances through distinct lists.

Devi et al. (2014) presented a generic anaphora resolution engine for Indian languages, employing Conditional Random Fields (CRF). However, most approaches for Hindi focus solely on pronominal resolution. Dakwale (2014) developed the first model to resolve nominal references, including pronominal ones, using a Rule-based approach, with reported MUC Precision, Recall, and F1-scores of 64%, 50%, and 56%, respectively. Sachan et al. (2015) developed a coreference resolution system for Hindi text based on an an active learning approach. The authors developed a method for resolving the in-document coreferences resolution that reduces the amount of human interference in this process. The performance of the coreference resolution system is better than Dakwale (2014) approach

Vasantlal (2017) recently proposed a hybrid sieve-based strategy for resolving pronouns and nominal references in Hindi, incorporating Paninian Dependency Grammar, POS labels, morphology, and linguistic resources like Hindi WordNet, DBpedia, Word2Vec, and GloVe. This method, however, relies on labeled datasets, with reported MUC Precision, Recall, and F1-scores of 79.53%, 63.7%, and 70%, respectively.

Ramrakhiyani et al. (2018) developed a Coreference Resolution system using Markov Logic Networks (MLN) to resolve actor mentions in Hindi narrative text. They evaluated their system on multiple datasets (Sardar, Plassey, Shivaji, Emergency, IIIT-H), reporting an average F1-measure of 70.46%, 64.91%, 68.98%, 63.12%, and 55.04%, respectively.

Mishra et al. (2024) presented TransMuCoRes, a translated dataset made with off-the-shelf tool for translation and word-alignment that is intended for Multilingual Coreference Resolution across 31 South Asian languages. On a test split of a manually annotated Hindi golden corpus, the top-performing model obtained LEA F1 64% and CoNLL F1 68%.

3 Coreference Resolution Model

This section explains how to resolve coreferences in Hindi text using the proposed approach. We employed the English Coreference Resolution approach outlined by Lee et al. (2018) for Coreference Resolution in Hindi text. We utilize a pre-trained Transformer-based Indic BERT (Kakwani et al., 2020) and MuRIL model (Khanuja et al., 2021; Devlin, 2018). The Coreference Resolution model for Hindi (DeepHCoref) consists of mention's span representation and a clustering step. The block diagram for DeepHCoref is shown in Figure 2.

3.1 Mention's Span Representation

We must create vector representations of words and spans. The following characteristics are used to construct word representations: (1) Word vectors derived from a pre-trained language model. (2) Word vectors regarding sentence context derived from a pre-trained language model. (3) Character-based word vectors. The vector representations of spans are created by combining all

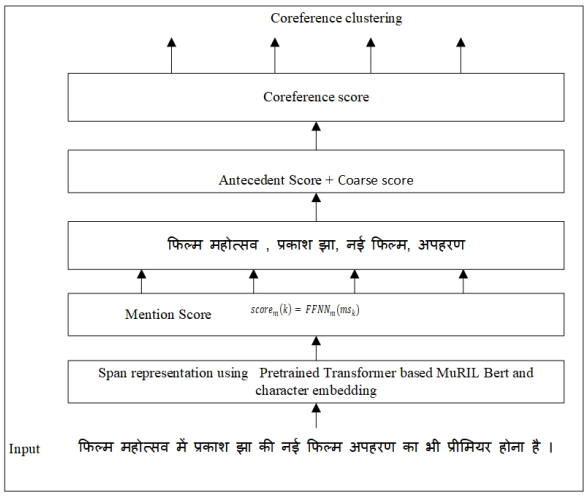


Figure 2: Block diagram of DeepHCoref.

these properties of words through concatenation operations, which are processed by recurrent layers with the help of the attention mechanism.

In our model, the span representation is created by employing pre-trained Indic BERT and MuRIL, whereas Lee et al. (2018) utilized ELMO embeddings. The authors used Bi-LSTM to get span representation, but we have utilized Bi-GRU for this purpose because we have a smaller training dataset, as described by Yang et al. (2020). They demonstrated that GRU is 29.29% faster than LSTM for small datasets and long texts in terms of training speed and performance.

First, we find the word embedding $\mathbf{vec}_i$ for each word w_i in a sentence from pre-trained Indic BERT, and then find the character embedding of the word through a Convolutional Neural Network (CNN). The concatenation of the word embedding with the character embedding is represented by $\mathbf{embed}_i$ for each word w_i, where $i = 1, 2, \ldots, W$, as shown in Figure 3. After this step, concatenated embedding $\mathbf{embed}_i$ is considered as input and given to a Bi-directional GRU (Bi-GRU) to generate word representations $\mathbf{x}_i$, where $i = 1, 2, \ldots, W$. The head-finding attention vector $\mathbf{hd}_k$ of a mention span is calculated as the weighted average of the mention's word representations as shown in equation 1.

$$\left.\begin{aligned} o_i &= FFNN_0(x_i) \\ att_{k,i} &= \frac{e^{o_i}}{\sum_{l=beg_k}^{end_k} e^{o_l}} \\ h_{dk} &= \sum_{l=beg_k}^{end_k} att_{k,i} \cdot x_i \end{aligned}\right\} \quad (1)$$

Where $\text{att}_{k,i}$ is the word-level attention parameter for the i-th word in the k-th mention, beg_k indicates the position of the starting word in the k-th mention, and end_k represents the ending position of a word. The mention's span representations ms_k are formed by combining $\mathbf{x}_i$ with head representations $\mathbf{hd}_k$, as shown in equation 2 and represented in Figure 4.

$$ms_k = [x_{beg_k}, x_{end_k}, h_{d_k}, \phi(k)] \quad (2)$$

Where $\phi(k)$ represents the mention feature embeddings. A feedforward neural network (FFNN) calculates the score of mention (sm_k) to identify the relevance of a candidate mention, as shown in equation 3.

$$score_m(k) = FFNN_m(ms_k) \quad (3)$$

3.2 Clustering Step

The next step is to link an antecedent for each possible mention. We calculate a lightweight mention pair score $\text{score}_{coarse}(k, n)$ between all relevant mention pairs (relevant mentions paired with all prior mentions) using a bilinear function, as shown in equation 4.

$$score_{coarse}(k, n) = ms_k^T W_{coarse} ms_n \quad (4)$$

These coarse scores are then used to select the best candidate antecedents. Next, we calculate a more accurate mention pair score, $\text{score}_{ant}(k, n)$, between the mention and its best antecedent candidate, as shown in equation 5.

$$\begin{aligned} \text{score}_{ant}(k, n) = \text{FFNN}_{ant}([ms_k, ms_n, \\ ms_k \odot ms_n, \phi(k, n)]) \end{aligned} \quad (5)$$

Where ms_k, ms_n are the antecedent and anaphora representations, and $\phi(k, n)$ is the feature vector of the distance between the mention pair. Finally, we compute the mention pairwise score $\text{score}(k, n)$, as shown in equation 6.

$$\text{score}(k, n) = \begin{cases} \text{score}_m(k) + \text{score}_m(n) \\ \quad + \text{score}_{ant}(k, n) \\ \quad + \text{score}_{coarse}(k, n), & k \neq \epsilon \\ 0, \quad k = \epsilon \end{cases} \quad (6)$$

Here, ϵ represents a fictitious antecedent in cases where the span is not a mention or when no antecedent exists in the candidate list. The antecedent with the highest $\text{score}(k, n)$ is predicted as the antecedent for each mention.

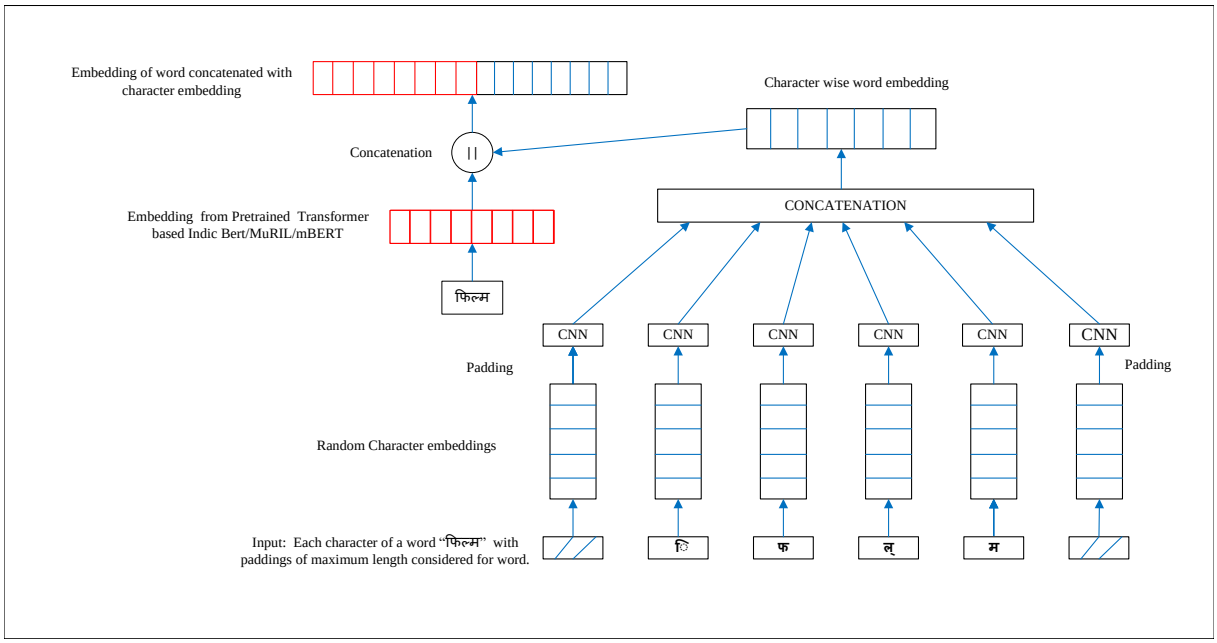


Figure 3: Concatenation of character embedding with Indic BERT /MuRIL/mBERT embedding.

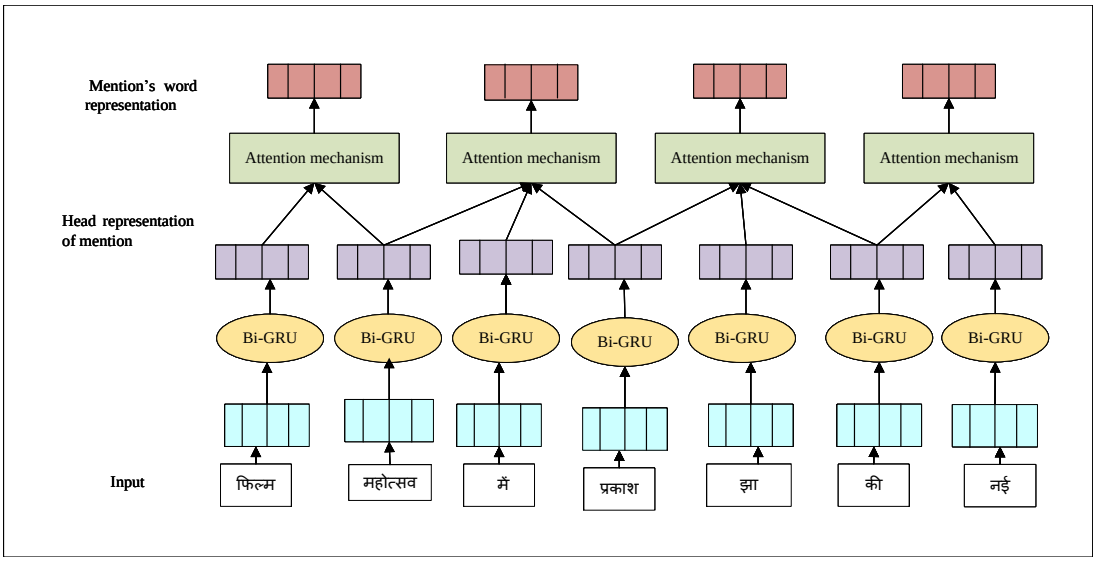


Figure 4: Mention span representation.

3.3 Data Preparation

Hindi, being a low-resource language, has limited training data available. We used coreference annotated data for the Hindi language Mujadia et al. (2016), which consists of 3.6K sentences and 78K tokens from news articles in the Hindi newspaper Amar Ujala, including news related to sports, politics, films, etc. The coreference annotated dataset created by the authors contains grammatical features such as number, gender, animacy features, dependency relations information, and chain of coreference and coreference relation types such as Part-of', 'Function-value pair' etc. Table 1 shows the corpus statistics. This dataset contains coreference chain which is created semi-automatically. We have assumed that the mentions and coreference chain annotated in this dataset are true. manual corrections were made as needed. We wrote a Python script to convert the dataset from SSF format Bharati et al. (2007) into JSON lines format, as shown in Figure 5.

Hindi Dataset	Size
# Documents	275
# Sentences	3.6K
# Tokens	78K

Table 1: Corpus statistics for Hindi dataset

{"clusters": [[[3,4],[18,18]], [[6,7],[8,8],[19,19], [25,26]]

"doc_key": "(nw/fullnews_id_250.txt); part 250",

"sentences": [[“फिल्म”, “महोत्सव”, “में”, “प्रकाश”, “झा”, “की”, “नई”, “फिल्म”, “अपहरण”, “का”, “भी”, “प्रीमियर” “होना”, “है”, “।”] [“गंगाजल”, “के”, “बाद”, “उसकी”, “यह”, “किसी”, “अलग”, “विषय”, “पर”, “बनी”, “दूसरी”, “फिल्म”, “है”, “।”]],

"speakers": [[“ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ” “ ”, “ ”, “ ”] [“ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ”, “ ”]]

}

Figure 5: Sample of text data in JSON lines format.

3.4 Mention Detection

We used an external mention detection system to detect mentions. Lata et al. (2022) reviewed mention detection algorithms and highlighted their importance in coreference resolution tasks. Aloraini et al. (2020) demonstrated that separate mention detection modules perform better than joint systems for coreference resolution. We used their approach for the detection of mentions, which trains the system end-to-end initially and gradually transitions to a pipeline-based approach. This technique mitigates the impact of false positive mentions and improves the performance of coreference resolution.

4 Experimental Setup and Evaluation

4.1 Experimental Setup

We used an NVIDIA 970GTX GPU and a 4.00 GHz Intel i7-4790 processor with 64GB RAM and TensorFlow backend support to train our models. In all experiments, the dataset is randomly split into training, development, and test sets. The training set is used for training the model, the development set for optimizing settings, and the test set for evaluating model performance.

Hyperparameters

The hyperparameter settings for the presented work are shown in Table 2. We used the default settings employed by Lee et al. Lee et al. (2018), and employed 300-dimensional fastText (IndicFT)[4] embeddings instead of GloVe/ELMo embeddings.

Parameter	Value	Parameter	Value
Word Embedding Dimension	300	Bi-GRU Dropout	0.5
Bi-GRU Size	200	Bi-GRU Layers	3
FFNN Layers	2	CNN Filter Widths	3,4,5
FFNN Layer Size	150	CNN Filter Size	50
FFNN Dropout	0.2	BERT Embedding Size	1024
Learning Rate	0.001	Decay Rate	0.999
Max Span Width	30	Max Antecedents	50
Mention Ratio	0.4	Optimizer	Adam

Table 2: Hyperparameter settings

Additionally, we employed three transformer-based BERT language models: MuRIL(Khanuja et al., 2021), Multilingual-BERT (mBERT) (Devlin et al., 2019), and IndicBERT (Kakwani et al., 2020).

4.2 Experimental Results

The system predicts mentions and coreferential mentions using the proposed approach. Results are evaluated using metrics such as MUC (Vilain et al., 1995), B-CUBE (Bagga and Baldwin, 1998), and CEAFϕ4 (Luo, 2005). The CoNLL-2012 scoring script (v8.01) (Pradhan et al., 2014) was used to evaluate the performance of our DeepHCoref system. As discussed in Section 3.4, We have applied an external mention detection module to detect the mentions. Table 3 shows the performance of the mention detection model with MuRIL, IndicBERT, and mBERT in both joint and separate settings in high recall setting. We have compared the joint model(in which we train both mention detection and Coreference Resolution simultaneously), and the separate model(in which we train mention

[4] https://indicnlp.ai4bharat.org/fasttext/

detection and Coreference Resolution separately) with different variants: Hindi Mention Detection with MuRIL (HMD – MuRIL), Hindi Mention Detection with IndicBERT (HMD – IndicBERT), and Hindi Mention Detection with mBERT (HMD-mBERT). The observation from the table is that the mention detection module, which is trained separately is consistently outperformed as compared to joint HMD. Table 4 shows the results of the Hindi Mention Detection (HMD) models which are not in the High Recall setting. It is observed that HMD-mBERT performed better than other variants.

Model	Joint model			Separate model		
	Recall	Precision	F-measure	Recall	Precision	F-measure
HMD - MuRIL	71.68	27.61	39.86	**74.18**	28.41	41.02
HMD - IndicBERT	74.53	28.71	41.45	**76.63**	29.31	42.40
HMD - mBERT	86.38	33.27	48.04	**89.38**	34.07	49.33

Table 3: Comparison of joint and separate Hindi Mention Detection (HMD) models

Model	**Recall**	**Precision**	**F-measure**
HMD - MuRIL	30.51	76.96	43.70
HMD - IndicBERT	36.23	80.74	50.01
HMD - mBERT	61.90	83.55	71.11

Table 4: Hindi Mention Detection (HMD) experimental results on test data

Table 5 shows the performance of the Coreference Resolution system on test data, which utilizes different BERT models (MuRIL, IndicBERT, and mBERT). We observe that the the best model variant combines mBERT (DeepHCoref + mBERT + HMD) with mBERT performs significantly better than those with IndicBERT and MuRIL.

We observed that IndicBERT's performance is limited, likely due to its smaller sequence length (128) and smaller training dataset compared to mBERT, which was trained with a sequence length of 512. However, the MuRIl was also trained on the sequence length, i-e., 512, same as mBERT, and trained explicitly for the Indian language. Surprisingly, the MuRIL model on our task performed lower than the IndicBERT and mBERT model on test set. The overall performance of our DeepHCoref + mBERT + HMD model is lower than the baseline rule-based model, likely due to the small dataset size.

Despite having a higher average CoNLL F1 measure score (67 vs. 55.47) than our model(DeepHCoref + mBERT + HMD), the wl-coref-xlmr model (Mishra et al., 2024) depends on a dependency parsing mechanism through the Stanza library(Qi et al., 2020). In certain languages or contexts where there is a dearth of training data or complex syntax, dependency parsers such as Stanza may parse sentences incorrectly due to their imperfection. The Coreference Resolution task may encounter difficulties if the dependency parse tree incorrectly recognizes heads or other syntactic relationships. On the other hand, our model does not rely on external syntactic parsers, which provides a simpler pipeline and eliminates the possibility of errors introduced by dependency parsers, especially in languages with limited resources. Further improvements could be achieved by training the model on a larger dataset. Moreover, creating a gold-standard Coreference Resolution dataset for Hindi would significantly enhance model performance. Currently, the available dataset is semi-automatically generated and does not explicitly label singleton mentions.

5 Conclusion and Future Scope

Coreference resolution is a crucial yet challenging problem in Natural Language Processing. In this research, we applied a state-of-the-art English coreference system to the Hindi language to enhance the Coreference Resolution task for Hindi. We presented a Hindi Coreference Resolution model, developed by integrating the multilingual language model MuRIL, which is specifically pre-trained for Indian languages/mBERT, along with CNN and Bi-GRU.

In this study, we also investigated the performance of the proposed system using IndicBERT and mBERT language models on the same dataset. The results show that the mBERT language model performs significantly better than both IndicBERT and MuRIL for the Hindi Coreference Resolution task. In future work, we will analyze the reasons behind the lower performance of our model with MuRIL-large.

The performance of the suggested model also demonstrates that the Hindi Coreference Resolution system, DeepHCoref, can be further improved by using a more extensive training dataset and a larger language model. Future research will explore in depth how the removal of singletons affects the Coreference Resolution system. Additionally, in this work, coreference is resolved within a single document; future studies can investigate the resolution of coreference problems

Model	MUC			B-CUBE			CEAFϕ4			Avg. (CoNLL)	
	R	P	F1	R	P	F1	R	P	F1	R	F1
Rule-based CR (Vasantlal, 2017)	63.7	79.53	70.00	-	-	-	-	-	-	-	-
wl-coref-xlmr (Mishra et al., 2024)	-	74	-	-	-	-	-	66	-	62	67
fast-coref-xlmr (Mishra et al., 2024)	-	45	-	-	-	-	-	35	-	33	38
DeepHCoref + MuRIL	23.79	63.57	34.62	16.33	59.17	25.60	17.61	44.86	25.29	28.50	28.50
DeepHCoref + IndicBERT	29.06	67.43	40.61	20.57	62.31	30.90	21.11	49.59	29.80	33.58	33.77
DeepHCoref + mBERT	53.39	72.74	61.85	43.04	66.86	52.37	40.67	61.75	48.56	54.17	54.17
DeepHCoref + mBERT + HMD	54.50	72.84	62.34	43.84	67.36	53.11	42.82	61.15	49.86	55.47	55.47

Table 5: Hindi Coreference Resolution results on the test set

across documents.

In this work, our model does not explicitly handle the zero mentions (pro-drop), Because there are no annotations for zero mentions (pro-drop) in the dataset we used. However for languages like Hindi, pro-drop must be addressed if Coreference Resolution is to be improved. We intend to investigate strategies for dealing with zero mentions in future work, such as utilizing syntactic features to infer implicit pronouns or adding pro-drop annotations to datasets. These modifications may improve the model's performance even more in low-resource languages As, Hindi dataset is not currently available in the CorefUD collection, despite notable progress in multilingual coreference resolution. Consequently, the Hindi coreference corpus made accessible byMujadia et al. (2016) is the foundation of our work. Our future research endeavors to investigate the integration of Hindi into multilingual datasets such as CorefUD.

Acknowledgements

The authors would like to thank all reviewers for reviewing this work, and providing very insightful comments which helped us to improve the quality of manuscript. The authors also would like to thank the authors Juntao Yu and Mujadia Vandan Vasantlal for providing support to understand after direct contact.

References

Abdulrahman Aloraini, Juntao Yu, and Massimo Poesio. 2020. Neural coreference resolution for arabic. In *Proceedings of the Third Workshop on Computational Models of Reference, Anaphora and Coreference*, pages 99–110.

Amit Bagga and Breck Baldwin. 1998. Algorithms for scoring coreference chains. In *The first international conference on language resources and evaluation workshop on linguistics coreference*, volume 1, pages 563–566. Citeseer.

Akshar Bharati, Rajeev Sangal, and Dipti M Sharma. 2007. Ssf: Shakti standard format guide. *Language Technologies Research Centre, International Institute of Information Technology, Hyderabad, India*, pages 1–25.

Anders Björkelund and Jonas Kuhn. 2014. Learning structured perceptrons for coreference resolution with latent antecedents and non-local features. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 47–57.

Kevin Clark and Christopher D Manning. 2016a. Deep reinforcement learning for mention-ranking coreference models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2256–2262.

Kevin Clark and Christopher D Manning. 2016b. Improving coreference resolution by learning entity-level distributed representations. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 643–653.

Praveen Dakwale. 2014. *Anaphora resolution in hindi*. Ph.D. thesis, PhD thesis, International Institute of Information Technology Hyderabad.

Hal Daumé III and Daniel Marcu. 2009. A large-scale exploration of effective global features for a joint entity detection and tracking model. *arXiv preprint arXiv:0907.0807*.

Sobha Lalitha Devi, Vijay Sundar Ram, and Pattabhi RK Rao. 2014. A generic anaphora resolution engine for indian languages. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1824–1833.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language*

Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Greg Durrett and Dan Klein. 2013. Easy victories and uphill battles in coreference resolution. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1971–1982.

Kamlesh Dutta, Nupur Prakash, and Saroj Kaushik. 2008. Resolving pronominal anaphora in hindi using hobbs algorithm. *Web Journal of Formal Computation and Cognitive Linguistics*, 1(10):5607–11.

Aria Haghighi and Dan Klein. 2009. Simple coreference resolution with rich syntactic and semantic features. In *Proceedings of the 2009 conference on empirical methods in natural language processing*, pages 1152–1161.

Jerry R Hobbs. 1978. Resolving pronoun references. *Lingua*, 44(4):311–338.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the association for computational linguistics*, 8:64–77.

Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel S Weld. 2019. Bert for coreference resolution: Baselines and analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5803–5808.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, NC Gokul, Avik Bhattacharyya, Mitesh M Khapra, and Pratyush Kumar. 2020. Indicnlpsuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for indian languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4948–4961.

Ben Kantor and Amir Globerson. 2019. Coreference resolution with entity equalization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 673–677.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, et al. 2021. Muril: Multilingual representations for indian languages. *arXiv preprint arXiv:2103.10730*.

Kusum Lata, Pardeep Singh, and Kamlesh Dutta. 2021. A comprehensive review on feature set used for anaphora resolution. *Artificial Intelligence Review*, 54:2917–3006.

Kusum Lata, Pardeep Singh, and Kamlesh Dutta. 2022. Mention detection in coreference resolution: survey. *Applied Intelligence*, 52(9):9816–9860.

Heeyoung Lee, Yves Peirsman, Angel Chang, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2011. Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Proceedings of the fifteenth conference on computational natural language learning: Shared task*, pages 28–34.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. *arXiv preprint arXiv:1707.07045*.

Kenton Lee, Luheng He, and Luke Zettlemoyer. 2018. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 687–692.

Xiaoqiang Luo. 2005. On coreference resolution performance metrics. In *Proceedings of human language technology conference and conference on empirical methods in natural language processing*, pages 25–32.

Ritwik Mishra, Pooja Desur, Rajiv Shah, and Ponnurangam Kumaraguru. 2024. Multilingual coreference resolution in low-resource south asian languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 11813–11826.

Vandan Mujadia, Palash Gupta, and Dipti Misra Sharma. 2016. Coreference annotation scheme and relation types for hindi. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 161–168.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Sameer Pradhan, Xiaoqiang Luo, Marta Recasens, Eduard Hovy, Vincent Ng, and Michael Strube. 2014. Scoring coreference partitions of predicted mentions: A reference implementation. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2014, page 30. NIH Public Access.

Rashrni Prasad and Michael Strube. 2000. Discourse salience and pronoun resolution in hindi.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. *arXiv preprint arXiv:2003.07082*.

Nitin Ramrakhiyani, Swapnil Hingmire, Sachin Pawar, Sangameshwar Patil, Girish Palshikar, Pushpak Bhattacharyya, and Vasudeva Verma. 2018. Resolving actor coreferences in hindi narrative text. In *Proceedings of the 15th International Conference on Natural Language Processing*, pages 50–58.

Mrinmaya Sachan, Eduard Hovy, and Eric P Xing. 2015. An active learning approach to coreference resolution. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Nikolaos Stylianou and Ioannis Vlahavas. 2021. A neural entity coreference resolution review. *Expert Systems with Applications*, 168:114466.

Rhea Sukthanker, Soujanya Poria, Erik Cambria, and Ramkumar Thirunavukarasu. 2020. Anaphora and coreference resolution: A review. *Information Fusion*, 59:139–162.

Bhargav Uppalapu and Dipti Misra Sharma. 2009. Pronoun resolution for hindi. In *7th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC 2009)*, pages 123–134.

Mujadia Vandan Vasantlal. 2017. *Capturing and resolving entities and their mentions in discourse*. Ph.D. thesis, Doctoral dissertation, International Institute of Information Technology ….

Marc Vilain, John D Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A model-theoretic coreference scoring scheme. In *Sixth Message Understanding Conference (MUC-6): Proceedings of a Conference Held in Columbia, Maryland, November 6-8, 1995*.

Sam Joshua Wiseman, Alexander Matthew Rush, Stuart Merrill Shieber, and Jason Weston. 2015. Learning anaphoricity and antecedent ranking features for coreference resolution. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics.

Wei Wu, Fei Wang, Arianna Yuan, Fei Wu, and Jiwei Li. 2020. Corefqa: Coreference resolution as query-based span prediction. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 6953–6963.

Shudong Yang, Xueying Yu, and Ying Zhou. 2020. Lstm and gru neural network performance comparison study: Taking yelp review dataset as an example. In *2020 International workshop on electronic communication and artificial intelligence (IWECAI)*, pages 98–101. IEEE.

Zdeněk Žabokrtskỳ, Miloslav Konopík, Anna Nedoluzhko, Michal Novák, Maciej Ogrodniczuk, Martin Popel, Ondřej Pražák, Jakub Sido, and Daniel Zeman. 2023. Findings of the second shared task on multilingual coreference resolution. In *Proceedings of the CRAC 2023 Shared Task on Multilingual Coreference Resolution*, pages 1–18.

Zdeněk Žabokrtskỳ, Miloslav Konopík, Anna Nedoluzhko, Michal Novák, Maciej Ogrodniczuk, Martin Popel, Ondřej Pražák, Jakub Sido, Daniel Zeman, and Yilun Zhu. 2022. Findings of the shared task on multilingual coreference resolution. In *Proceedings of the CRAC 2022 Shared Task on Multilingual Coreference Resolution*, pages 1–17.