Extracting Message Sequence Charts from Hindi Narrative Text

Swapnil Hingmire **Nitin Ramrakhiyani** **Avinash Kumar Singh**[*]
Sangameshwar Patil **Girish K. Palshikar**
{swapnil.hingmire,nitin.ramrakhiyani,singh.avinash3}@tcs.com
{sangameshwar.patil,gk.palshikar}@tcs.com
TRDDC, TCS Research and Innovation, India

Pushpak Bhattacharyya
pb@cse.iitb.ac.in
IIT Patna, India

Vasudeva Varma
vv@iiit.ac.in
IIIT Hyderabad, India

Abstract

In this paper, we propose the use of Message Sequence Charts (MSC) as a representation for visualizing narrative text in Hindi. An MSC is a formal representation allowing the depiction of actors and interactions among these actors in a scenario, apart from supporting a rich framework for formal inference. We propose an approach to extract MSC actors and interactions from a Hindi narrative. As a part of the approach, we enrich an existing event annotation scheme where we provide guidelines for annotation of the mood of events (realis vs irrealis) and guidelines for annotation of event arguments. We report performance on multiple evaluation criteria by experimenting with Hindi narratives from Indian History. Though Hindi is the fourth most-spoken first language in the world, from the NLP perspective it has comparatively lesser resources than English. Moreover, there is relatively less work in the context of event processing in Hindi. Hence, we believe that this work is among the initial works for Hindi event processing.

1 Introduction

Narratives are used to communicate complex ideas, detailed accounts of complex events or arguments about one's beliefs (Valls Vargas, 2017). Moreover, a narrative is a powerful tool not just from entertainment perspective but is one of the core component of human memory, knowledge and intelligence (Schank and Morson, 1995). Narrative texts are common in History where they mostly describe events that have happened in the past. Narrative text can also be routinely seen in news articles reporting events that have happened (or about to happen) in various political, corporate and social walks of a country. For multiple applications in text analysis, it becomes important to understand and analyse such narratives. A narrative has two key aspects, *plot* also referred to as *story* and *discourse*, the way in which the plot is described (Chatman, 1975). In this paper we focus on visualization of the plot aspect of a Hindi narrative.

किसानों ने अंग्रेज सरकार से भारी कर में छूट की मांग की। (The farmers demanded a waiver in the heavy taxes from the British Government.)
जब यह स्वीकार नहीं किया गया तो सरदार पटेल और गांधीजी ने किसानों का नेतऋत्व किया और उन्हे कर न देने के लिये प्रेरित किया।(When the British government didn't agree, Sardar Patel and Gandhi led them and motivated them to not pay the taxes.)
अन्त में सरकार झुकी और उस वर्ष करों में राहत दी गयी। (In the end, the government agreed and that year, waivers were provided in the tax.)

Table 1: Example Text Narrative

Hindi is an Indo-Aryan language spoken by around 300 million people in India. Additionally, Hindi is the fourth most-spoken first language in the world[1]. In comparison to English, Hindi has different linguistic characteristics leading to a different set of NLP challenges. First of all, Hindi is a Subject-Object-Verb (SOV) language with relatively free word order, as against the SVO order in English. Secondly, Hindi does not have high accuracy NLP toolkits such as Stanford CoreNLP (Manning et al., 2014). In this paper, we make one of the first attempts to facilitate event processing in Hindi by proposing annotation guidelines for events as well as *their arguments*.

We propose to represent a Hindi narrative using Message Sequence Charts (MSC) (Rudolph

[*] Work done during internship at TCS Pune.

[1] https://en.wikipedia.org/wiki/Hindi

Proceedings of the 1st Joint Workshop on Narrative Understanding, Storylines, and Events, pages 87–96
July 9, 2020.

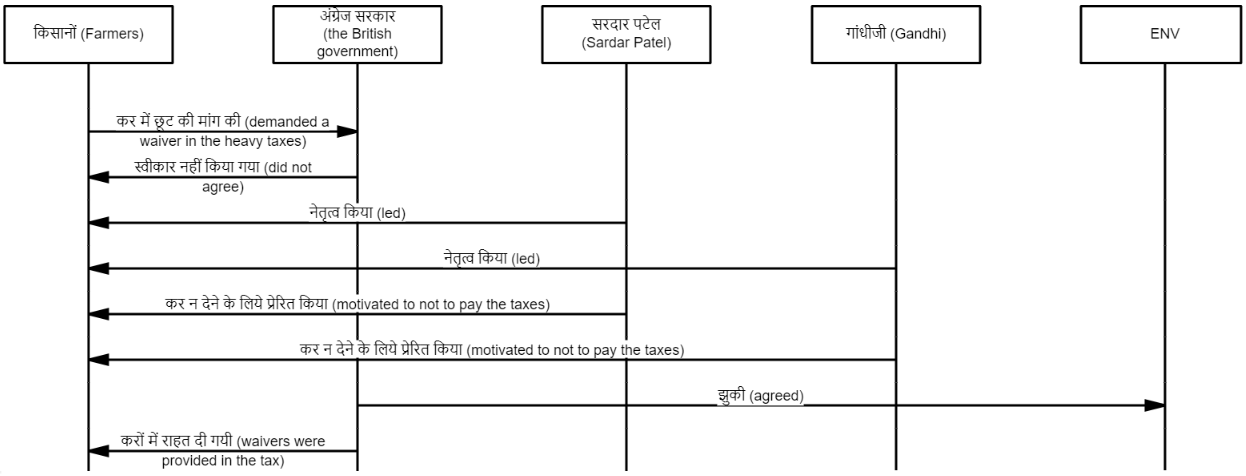
किसानों (Farmers)
अंग्रेज सरकार (the British government)
सरदार पटेल (Sardar Patel)
गांधीजी (Gandhi)
ENV
कर में छूट की मांग की (demanded a waiver in the heavy taxes)
स्वीकार नहीं किया गया (did not agree)
नेतृत्व किया (led)
नेतृत्व किया (led)
कर न देने के लिये प्रेरित किया (motivated to not to pay the taxes)
कर न देने के लिये प्रेरित किया (motivated to not to pay the taxes)
झुकी (agreed)
करों में राहत दी गयी (waivers were provided in the tax)

Figure 1: MSC of example text narrative described in Table 1

et al., 1996). MSC is a widely used notation to depict interactions among components of communicating devices. In an MSC, a vertical line represents a component and horizontal arrows indicate message from one component to another. We assume actors (or entities) in a narrative as components and message among the components as interactions among the corresponding actors. As an example, consider the text narrative from the Indian Independence Movement in Table 1 and the corresponding MSC in Figure 1.

We also address the problem of automatically extracting the MSC representation from a given narrative text. MSC extends the notion of a single event-timeline (Bedi et al., 2017) for a narrative by providing a timeline per actor (entity of interest). MSC representation captures all the actors and interactions in an easy to visualize manner and hence make the text more comprehensible. Further, the representation's support for inference mechanisms opens up possibilities of tackling natural language understanding problems like text comprehension and question answering. Our work is similar to (Palshikar et al., 2019) to represent a English narrative using MSC. However, due to intricacies of events in Hindi language, their approach cannot be used for construction of MSC of a Hindi narrative.

The key contributions of the paper are (i) we extend an existing scheme of annotation of events (Goud et al., 2019); we provide guidelines for annotation of mood of events (realis vs irrealis) and guidelines for annotation of event arguments, (ii) we propose an approach to identify event predicates and their arguments, (iii) MSC based visualization of actors and interactions, (iv) we report performance on multiple evaluation criteria by experimenting with four Hindi narratives from Indian History.

2 Key Annotation and Extraction Challenges for Events in Hindi

Annotation and extraction of events and their arguments from English texts is a challenging task (Mitamura et al., 2015). In case of Hindi there are more challenges as compared to English. Following are the key challenges we observed while processing Hindi narratives.

I. Absence of annotation guidelines and labelled data similar to ACE: Only few attempts have been made to define comprehensive event annotation guidelines for Hindi. (Goud et al., 2019; Goel et al., 2020) propose a set of guidelines for annotation of event mentions in Hindi. However, they do not consider arguments of events, which are vital for narrative processing.

II. Annotation of events with Light Verb Constructs (LVCs): LVCs are formed from a commonly used verb and usually a noun phrase (NP) in its direct object position, such as *have a look* or *take an action*. For example, in the sentence राम ने किताब दी ('Ram ne Kitab di'; Ram gave the book), दी (di;*give*) is the event, however in the sentence राम ने किताब खो दी ('Ram ne Kitab kho di'; *Ram lost the book.*), खो दी('kho di'; *lost*) is the event. Similarly, in the sentence राम से किताब गुम हो गई ('Ram se kitab gum ho gayee'; *Ram lost the book*), गुम हो गई ('gum ho gayee'; *lost*) is the event. As compared to English, LVCs are more common in Hindi, have different characteristics and are used as a preferred method for introducing new predicates into the language (Vaidya et al., 2016). A state-of-the-art approach (Chen et al., 2015) propose a supervised approach to identify LVCs in English using resources like PropBank, the OntoNotes sense groupings, WordNet and the British National Corpus. However, it is difficult to extend this approach for Hindi as it would require extensive efforts to create labelled data based on such resources for Hindi.

III. Annotation of nominal events: An event is a nominal event if it is described by a noun. Annotation of nominal events is a challenging task, as eventiveness of a noun depends on the context in which it is used. In this paper, we restrict our focus to events pivoted around verbs in the past tense as interactions. We plan to explore nominal events in the future.

3 Proposed Annotation Guidelines and Approach

As highlighted earlier, an MSC consists of a set of actors and interactions among them. The proposed approach is accordingly divided into two major stages: (1) Extraction of actors and their coreference resolution; (2) Extraction of interactions with their sender and receiver arguments.

3.1 Actor Extraction and Coreference Resolution

Since we are focusing on narratives of historical events, we assume the actors to be entities of type PERSON, LOCATION and ORGANIZATION. Further, we are interested in identifying all types of actor mentions - proper nouns, common nouns and pronouns. To extract proper noun mentions, we perform Named Entity Recognition of the text and mark all noun phrases of type PERSON, LOCATION and ORGANIZATION as actors. Since Hindi is a relatively resource-poor language and it is very costly to generate necessary labeled data to train a supervised model, we propose an unsupervised technique to identify actors based on Wordnet. Given a noun which is the head of a noun phrase, we query its hypernym hierarchy to check if specific senses of स्थान (sthaan; *place*), क्षेत्र (kshetra; *region*), भू-भाग (bhu-bhaag; *geographical area*), व्यक्ति (vyakti; *person*), मानव (maanav; *person*) and समुदाय (samudaay; *group*) are found. If so, we tag the type of the respective noun phrase corresponding to the hypernym found. As pronouns are a closed set of words, we use a manually prepared list of pronouns and corresponding types to identify all pronoun mentions of the actors. It is ensured that the list does not include any demonstrative pronouns (such as यह, वह, उन, उस). Each pronoun in the text is checked against the list and marked as an actor, if found.

As the actor mentions are to be visualized on a MSC, it is not sufficient to depict only head words of actor phrases. For example, सरदार पटेल (*Sardar Patel* is more informative as well as describes the complete entity than just the headword पटेल (*Patel*). Hence, we propose a simple dependency parse based approach to identify the complete noun phrase given the head word. We append to the head word all nouns which are dependent on it through the *compound* dependency. For example, the word महात्मा (*Mahatma*) is dependent on the head word गांधी (*Gandhi*) through the *compound* dependency and hence, the complete phrase becomes महात्मा गांधी (*Mahatma Gandhi*). We also append any adjectives/quantifiers which modify the head word using the *amod* dependency. For example, प्रथम प्रधानमंत्री (pratham pradhanmantri; *first prime minister*) is formed as a complete phrase where प्रथम (pratham; *first*) modifies प्रधानमंत्री (pradhanmantri; *prime minister*) using the *amod* dependency. A complex construct observed is a noun dependent on the head word with a *nmod* dependency with no verbs or prepositions occurring between the two. Such nouns are also considered part of the complete phrase and are handled through this approach. For example, in the phrase स्वतंत्रता संग्राम सेनानी (swatantrata sangram senani; *freedom fighter*), the word स्वतंत्रता (swatantrata; *independence*) is dependent on the head word सेनानी (senani; *soldier*) through the *nmod* dependency with only a noun संग्राम (sangram; *struggle*) appearing in between.

For coreference resolution, we employ the technique proposed in (Ramrakhiyani et al., 2018). The authors assume gold actor mentions as input and predict coreferences between the actor mentions. They apply a set of linguistically motivated rules coded in a Markov Logic Network (MLN) based framework to perform the coreference resolution. In this paper, we develop an MLN with the stated rules and input it the extracted actor mentions to perform the coreference resolution. We also annotate the first observed mention in each mention cluster as the canonical actor mention for that coreference group. The canonical actor mention is used for its depiction on the output MSC.

3.2 Interaction Extraction

An interaction in the MSC notation is composed of the central event and its sender and receiver actor arguments. To extract interactions it hence becomes important to identify the meaningful events and their actor arguments. In this paper, we focus on events which are primarily described by verbs. We define such verbal events based on linguistic properties of Hindi verbs (Vaidya, 2015). We also propose a set of annotation guidelines to mark such verbal events and their arguments.

3.2.1 Verbal Events

A verbal event can be of two types: 1) simple predicate where a verb triggers the event. For example, in the sentence, राम ने आम **खाया** (Ram ne aam khaya; *Ram ate a mango*.), the verb **खाया** is a simple predicate, 2) complex predicate where an event is triggered by a verb and an additional element of type *verb* or *adjective* or *adverb* or borrowed English verb or *noun* (Vaidya, 2015). A complex predicate further has two important types: 1) *conjunct predicate* where the additional element is of type *noun*. For example, in the sentence, २०१७ में राम का **जन्म** **हुआ** (2017 me Ram ka janma hua; *Ram was born in 2017*.) the verbal predicate हुआ triggers an event with the additional noun element **जन्म**, 2) *compound predicate* where the additional element is of type *verb*. For example, in the sentence, राम ने किताब **खो** **दी** (Ram ne kitab kho di; *Ram lost the book*.), the verbal predicate दी triggers an event with the additional verb element **खो**.

3.2.2 Annotation of Events

The proposed annotation guidelines for marking verbal events are described as follows:

I. In this paper, we restrict our scope to events represented using verbs. For example, in the following sentence महोत्सव (mahotsav; *festival*) is not annotated as an event:
फिल्म महोत्सव के दौरान पुराने अभिनेता पधारे (film mahotsav ke dauran purane abhineta padhare; *senior actors arrived during the film festival*)

II. The head verb of an event predicate is tagged as PIVOT. If an event predicate is a *conjunct predicate*, its *noun* element is tagged as P-CONJ. In case of a *compound predicate*, the *verb* element of the head verb is tagged as P-COMP. Following are the examples of *conjunct predicate* and *compound predicate* annotated as per this guideline:

Conjunct predicate:
उनकी $\underbrace{\text{शिक्षा}}_{\text{P-CONJ}}$ मुख्यतः स्वाध्याय से ही $\underbrace{\text{हुई}}_{\text{PIVOT}}$
(unki shiksha mukhtah swadhyay se hui;*His education happened mainly through self-learning*)

Compound predicate:
उन्हे ग्रह मंत्री का कार्य $\underbrace{\text{सौंपा}}_{\text{PIVOT}}$ $\underbrace{\text{गया}}_{\text{P-COMP}}$
(unhe grha mantri ka karya saupa gaya;*He was made in-charge of the home ministry*)

III. Based on the guidelines in (Mitamura et al., 2015), events with realis mood are considered valid events. An event is in realis mood, if it has explicitly happened in the past. On the other hand, if an event has irrealis mood then we can not say whether the event has actually happened or not. Following are examples of sentences with realis and irrealis mood events:

- Realis: लंदन जाकर उन्होंने बैरिस्टरी की पढाई की
(landan jakar unhone baristari ki padhai ki; *He went to London and studied law*)
- Irrealis: यदि सरदार कुछ वर्ष और जीवित रहते तो भारत का कायाकल्प हो जाता
(yadi sardar kuch varsh aur jeewit rehte to pure bharat ka kayakalp ho jata; *If Sardar remained alive for a few more years, India would have been transformed.*)

IV. Only punctual events are annotated as events. An event is punctual if it "does not have a transitional phase between its start and end point" (Kay and Aylett, 1996). This implies that a process in continuation is not punctual and hence not marked as a valid event.

- Punctual event: किसानों ने अंग्रेज सरकार से भारी कर में छूट की मांग की
(kisano ne angrejh sarkar se bhari kar me chut ki mang ki; *The farmers demanded the British government, a waiver in the heavy taxes*)
- Non-punctual event: गुजरात का खेडा खंड उन दिनों सूखे की चपेट में था
(gujrat ka kheda khand un dino sukhe ki

Algorithm 1: Identification of Event Predicates

```
Input: Sentence s
Result: $E$ : List of event predicate tuples:
        $< PIVOT, P\text{-}CONJ, P\text{-}COMP >$ in
        sentence s
$E := \emptyset$ // list of event predicates
$S_d := GetDependencyTree(s)$;
$root := S_d.root$ // get the root of the
    dependency tree
if root is a past tense verb then
    $e :=<>$; $e.PIVOT = root$;
    $E := E \cup e$;
    foreach child $c \in S_d.root.children$ do
        if
          $dep\_rel(c, root) \in [advcl, conj, acl : relcl]$
          then
            $e :=<>$; $e.PIVOT = c$;
            $E := E \cup e$
foreach event predicate $e \in E$ do
    foreach child $c \in S_d.e.children$ do
        if $dep\_rel(c, e) == compound$ and c is a
          noun then
            $e.P\text{-}CONJ = c$
        else if $dep\_rel(c, e) \in [aux, auxpass]$ and c
          is an auxiliary verb then
            $e.P\text{-}COMP = c$
return $E$
```

chapet me tha; *Those days Gujarat's Kheda district was affected by drought.*)

- Verbs appearing in the authors' opinions are not considered valid events. For example, the verbs in the quoted part जाएंगे (jayenge;will go) and है (hai) of the following sentence are ignored:
  सरदार पटेल ने केवल इतना कहा “क्या हम गोवा जाएंगे, केवल दो घंटे की बात है”
  (sardar patel ne kewal itna kaha “kya hum goa jayenge, kewal do ghante ki baat hai”; *Sardar Patel only said “Will we go to Goa, its a matter of only two hours”*.)

3.2.3 Annotation of Event Arguments

We consider an actor who initiates or triggers an event as the sender of the event. All other actors that participate in the event are considered as receivers of the event. For example, in the sentence राम ने राजू को लन्दन भेजा (ram ne raju ko london bheja; *Ram sent Raju to London*), राम is the sender of the event भेजा, while राजू and लन्दन are its receivers.

Algorithm 2: Identification of Interactions

```
Input: Sentence s, $E$ = List of event predicate tuples:
       $< PIVOT, P - CONJ, P - COMP >$ in
       sentence s, $A$ = List of actors in s
Result: $I$ = List of tuples:
        $< SENDER, PIVOT, RECEIVER >$
$I := \emptyset$ // list of interactions
$S_d := GetDependencyTree(s)$;
$root := S_d.root$ // get the root of the
    dependency tree
$rootSender := NULL$;
$rootCONJ := NULL$;
foreach Event $e \in E$ do
    $p := e.PIVOT$ $n := e.P\text{-}CONJ$
      $i :=< NULL, p, NULL >$;
    foreach child $c \in S_d.p.children$ do
        if $dep\_rel(c, p) == nsubj$ and $c \in A$ then
            $i.SENDER := c$;
            if p is root then
                $rootSender := c$;
                $rootCONJ := n$
        else if $dep\_rel(c, p) \in [dobj, nmod]$ and
          $c \in A$ then
            $i.RECEIVER := c$
    if $i.SENDER$ is $NULL$ or $i.RECEIVER$ is
      $NULL$ then
        foreach child $c \in S_d.n.children$ do
            if $i.SENDER$ is $NULL$ and
              $dep\_rel(c, n) == nsubj$ and $c \in A$
              then
                $i.SENDER := c$;
                if $n == rootCONJ$ then
                    $rootSender := c$
            else if $i.RECEIVER$ is $NULL$ and
              $dep\_rel(c, n) \in [dobj, nmod]$ and
              $c \in A$ then
                $i.RECEIVER := c$
    if $i.SENDER$ is $NULL$ and e is not root then
        $i.SENDER := rootSender$
    $I := I \cup i$
return $I$
```

3.3 Approach to Identify Verbal Events

In this paper, we propose an approach to identify the above described verbal events and their arguments. Our proposed approach makes use of POS tagging and dependency parsing.

Algorithm 1 shows our approach to identify event predicates and their compound and conjunct predicates (if available) from a sentence.

Figure 2 shows an illustration of our approach on an example sentence. The verb हुए (hue; *became*) is the root of the dependency tree of the sentence. We can determine the tense of हुए based on its child थे (thhe) with the dependency relation *aux*. As हुए is the root of the sentence with past tense, we identify it as the PIVOT of an event.

In the dependency tree of this sentence, the word नाराज (naraj; *unhappy*) is a *noun* child of हुए with dependency relation *compound* and hence, we identify नाराज as P-CONJ of हुए. Similarly, we identify थे as its P-COMP.

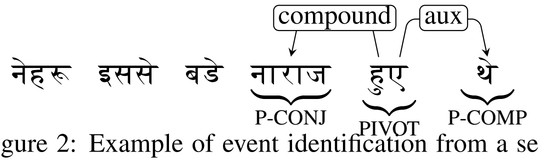

Figure 2: Example of event identification from a sentence (only relevant dependencies are shown for the sake of brevity)

In order to filter out an irrealis or non-punctual event, we check its respective P-COMP against a manually curated list of verbs such as जाता ,जाया, रही, रहे, चाहे, चाहती, होना, होती indicating either continuity or uncertainty of events. We also ignore events appearing in quotes as they are likely to be authors' opinions.

3.4 Approach to Identify Event Arguments

Algorithm 2 discusses our event argument identification approach leading to formation of the interactions to be shown on an MSC.

Figure 3 shows an illustration of our approach on an example sentence. For the event pivot लिखा (likha; wrote), words गांधी (Gandhi) and पटेल (Patel) are its dependency children with relations *nsubj* and *dobj* respectively. They are also present in the list of extracted actors and hence are designated as sender and receiver of this event.

In Figure 4, the word छूट (chhuut; waiver) is dependent on the P-CONJ - मांग (maang; demand) through the dependency relation *nmod*, but is not a valid actor and hence not a valid receiver.

4 Experimentation Details

4.1 Datasets

We carry out our experiments on the four text narratives from Indian History, contributed by (Ramrakhiyani et al., 2018). We obtain the dataset text and gold actor annotations and we carry out the event annotations for these datasets with the help of three annotators. The statistics about the datasets are described in Table 2.

	sardar	emergency	plassey	shivaji
# Sentences	90	56	74	71
# Words	1661	1373	1361	1293
# Actors	115	78	79	112
# Interactions	88	74	57	69

Table 2: Dataset Statistics

4.2 Experimental Setup

We used the Google SyntaxNet[2] for dependency parsing the Hindi sentences. SyntaxNet is a TensorFlow toolkit for deep learning powered natural language understanding developed at Google. The *Parsey Universal* component of SyntaxNet supports NLP preprocessing tasks such as POS tagging, morphological analysis and dependency parsing for 40 different languages including Hindi.

We employed two different NER approaches proposed for Hindi and consider a word as part of a named entity if either or both of them identify it as a named entity. One of the approaches is Polyglot, proposed in (Al-Rfou et al., 2013). It is based on using language agnostic techniques involving Wikipedia and Freebase and no human annotated NER training data. The second approach is proposed by (Murthy et al., 2019) and is based on a supervised deep learning architecture for NER in Hindi. To access the Hindi WordNet (Narayan et al., 2002), we use the pyiwn toolkit (Panjwani et al., 2018) which is a python API to access Indian language WordNets.

We use the *Sardar* dataset as a training set to iteratively revise and improve the extraction algorithms while keeping the other datasets unseen.

4.3 Evaluation and Results

The approach comprises of multiple facets like actor identification, actor coreference resolution, event extraction and event argument finding. To assess each of these facets, we carry out evaluation of the proposed approach at multiple levels.

As the first level, we check the performance of actor identification and coreference resolution. If an actor predicted by the approach is present in the gold standard, it is marked as a true positive. False positives and false negatives are computed accordingly. We report the F1 scores for actor mention identification for each dataset in Table 3. At this level, we also report the MUC (Vilain et al., 1995), the B^3 (Bagga and Baldwin, 1998) and the

[2] https://github.com/tensorflow/models/tree/master/research/syntaxnet

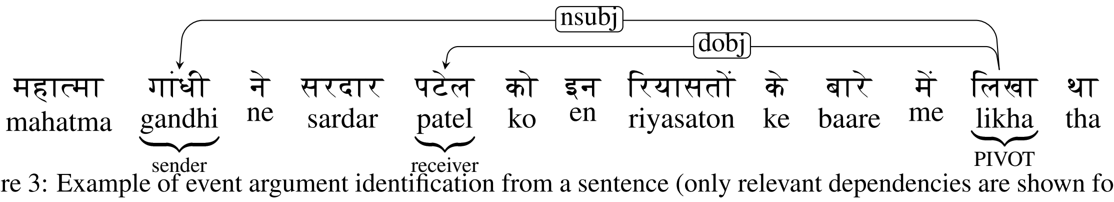

Figure 3: Example of event argument identification from a sentence (only relevant dependencies are shown for the sake of brevity)

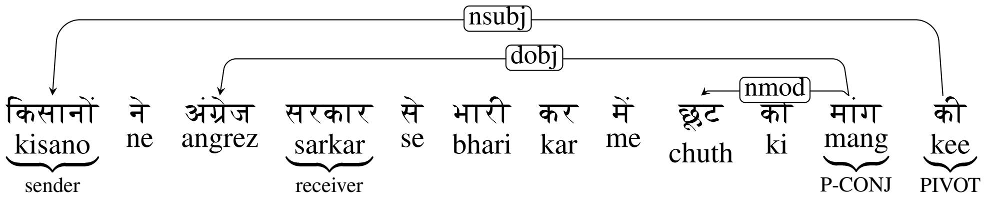

Figure 4: Example of valid and invalid arguments of an event predicate (only relevant dependencies are shown for the sake of brevity)

CEAFe (Luo, 2005) scores to measure the coreference resolution performance.

Dataset	Participant mentions	Canonical mentions & aliases		
		BCUB	MUC	CEAFe
sardar	68.2	45.7	51.0	48.6
emergency	71.0	41.2	44.1	43.1
plassey	68.5	40.6	50.4	34.6
shivaji	62.1	42.8	48.7	38.9

Table 3: Performance of Actor Identification and Coreference Resolution (F_1 metric in %).

At the next level, we check the correctness of events which are basis for interactions. If the predicted pivot is present in the gold standard, it is counted as a true positive and the false positives and false negatives are computed accordingly. In a similar way, true positives, false positives and false negatives are computed for P-CONJ and P-COMP parts. We report F1 scores for each of PIVOT, P-CONJ and P-COMP separately and the three combined i.e. the complete event, in Table 4.

At the final level, we check correctness of the extracted interactions which involves checking the correctness of both the event and its actor arguments. In Table 5, we report F1 scores for three settings namely Sender+Pivot (combination of both event pivot and corresponding sender should be correct), Receiver+Pivot (combination of both event pivot and corresponding receiver should be correct), Complete interaction (complete combination of event pivot, corresponding sender and corresponding receiver should be correct).

4.4 Analysis

From the low to moderate results, it can be observed that each stage of the approach is challenging. It is important to note here that this is a pipeline based approach and errors propagate from one stage to the next. A considerable number of errors is also attributed to the performance of linguistic tools such as SyntaxNet and Hindi NER.

We now discuss some error cases for the actor extraction stage. In the sentence इसके चलते कई अवसरों पर दोनो ने ही अपने पद का त्याग करने की धमकी दे दी थी (iske chalte kai avsaro par dono ne hi apne pad ka tyaag karne ki dhamki de di thi; *Meanwhile, on multiple occasions both had threatened to resign from their posts*), the gold actor दोनो (dono; both) is a noun but is not tagged as a valid actor. This is because none of the hypernyms of दोनो carry a PERSON sense and hence, our WordNet based approach fails to identify दोनो as a valid actor. This example highlights the need of a more richer knowledge resource that can disambiguate words like दोनो as an actor from its quantifier sense. Another error case is regarding pronouns like इसको (isko; this/him/her) which can be used to refer to events or actors. Pronouns like these are present in our pronoun list because of their actor referring property, but at places in the text when it appears referring a non-actor it is still considered as an actor leading to a false positive.

We also discuss error cases for the interaction extraction stage. In the sentence लन्दन जाकर उन्होंने बैरिस्टर की पढाई की और वापस आकर अहमदाबाद में वकालत करने लगे (landan jakar unhone baristri ki padhai ki aur wapas aakar ahmedabad me vakalat karne lage; *He went to London, studied law, came back and started practising law in Ahmedabad.*), the PIVOT आकर (aakar; came) does not find the right sender argument उन्होंने (unhone; he) using the dependency

	Sardar Patel	Emergency	Battle of Plassey	Shivaji Maharaj
PIVOT	0.82	0.82	0.64	0.79
P-CONJ	0.72	0.69	0.55	0.53
P-COMP	0.83	0.84	0.64	0.81
PIVOT+P-CONJ+P-COMP	0.66	0.62	0.47	0.46

Table 4: F1 scores - Event Evaluation

	Sardar Patel	Emergency	Battle of Plassey	Shivaji Maharaj
Sender+PIVOT	0.54	0.43	0.43	0.55
Receiver+PIVOT	0.50	0.39	0.35	0.48
Complete Interaction	0.38	0.30	0.25	0.41

Table 5: F1 scores - Interaction Evaluation

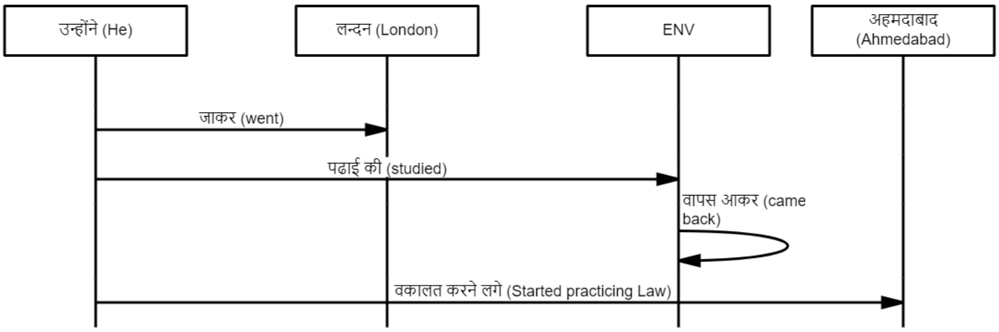

Figure 5: Predicted MSC of the example sentence "लन्दन जाकर उन्होंने बैरिस्टर की पढाई की और वापस आकर अहमदाबाद में वकालत करने लगे" (*He went to London, studied law, came back and started practising law in Ahmedabad.*)

parse based argument finding approach. This highlights the importance of more robust approach to identify arguments of events.

As part of the analysis, we show the visual MSC based depiction of the earlier sentence लन्दन जाकर उन्होंने बैरिस्टर की पढाई की और वापस आकर अहमदाबाद में वकालत करने लगे in Figure 5. As pointed out, the event आकर does not form a valid interaction because of absence of correct arguments.

5 Related Work

Automatic extraction of MSCs from narratives is studied for English language. Recently, (Palshikar et al., 2019) proposed a semantic role labelling and dependency parsing based approach to extract messages discussed in a narrative. They further use the Document Level Time-anchors algorithm for temporal ordering of extracted messages.

Several researchers such as (Rao and Devi, 2017, 2018) have focused on extraction of events from Hindi. However, these works are focused on news and social media texts. (Kuila et al., 2018) propose a neural network based approach for event extraction. However, the approach is supervised and needs labelled data. On the other hand, our approach does not need labelled data.

6 Conclusions

In this paper we proposed the use of a knowledge representation known as Message Sequence Charts (MSC) for visualizing Hindi narratives by identifying and depicting the multiple actors and interactions involved. As per our knowledge this is the first attempt to visualize Hindi narratives and represent them formally. We extend a set of annotation guidelines for marking events and their arguments pivoted on verbs. We also propose linguistic knowledge based approach for actor and interaction identification. We report results on four state-of-the-art Hindi narrative datasets and present a brief analysis of the approaches. As future work, we would like to extend this work on lines of Semantic Role Labelling (SRL) for Hindi including nominal predicates and more arguments. We also aim to work on the temporal ordering aspect of Hindi narratives which would allow us to depict interactions in the true chronological order.

References

Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed Word Representations for Multilingual NLP. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 183–192, Sofia, Bulgaria. Association for Computational Linguistics.

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. Granada.

Harsimran Bedi, Sangameshwar Patil, Swapnil Hingmire, and Girish Keshav Palshikar. 2017. Event Timeline Generation from History Textbooks. In *Proceedings of the 4th Workshop on Natural Language Processing Techniques for Educational Applications, NLP-TEA@IJCNLP 2017, Taipei, Taiwan, December 1, 2017*, pages 69–77.

Seymour Chatman. 1975. Towards a Theory of Narrative. *New literary history*, 6(2):295–318.

Wei-Te Chen, Claire Bonial, and Martha Palmer. 2015. English Light Verb Construction Identification using Lexical Knowledge. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Pranav Goel, Alok Debnath, Suhan Prabhu, and Manish Shrivastava. 2020. Hindi TimeBank: An ISO-TimeML Annotated Reference Corpus. In *Workshop on Interoperable Semantic Annotation (ISA-16)*.

Jaipal Singh Goud, Pranav Goel, Alok Debnath, Suhan Prabhu, and Manish Shrivastava. 2019. A Semantico-Syntactic Approach to Event-Mention Detection and Extraction In Hindi. In *Proceedings 15th Joint ACL-ISO Workshop on Interoperable Semantic Annotation*, pages 63–76.

Roderick Kay and Ruth Aylett. 1996. Transitivity and Foregrounding in News Articles: Experiments in Information Retrieval and Automatic Summarising. In *34th Annual Meeting of the Association for Computational Linguistics, 24-27 June 1996, University of California, Santa Cruz, California, USA, Proceedings.*, pages 369–371.

Alapan Kuila, Sarath chandra Bussa, and Sudeshna Sarkar. 2018. A Neural Network based Event Extraction System for Indian Languages. In *FIRE (Working Notes)*, pages 291–301.

Xiaoqiang Luo. 2005. On Coreference Resolution Performance Metrics. In *HLT/EMNLP 2005*, pages 25–32. Association for Computational Linguistics.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Prismatic Inc, Steven J. Bethard, and David Mcclosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *In ACL, System Demonstrations*.

Teruko Mitamura, Zhengzhong Liu, and Eduard H. Hovy. 2015. Overview of TAC KBP 2015 Event Nugget Track. In *Proceedings of the 2015 Text Analysis Conference, TAC 2015, Gaithersburg, Maryland, USA, November 16-17, 2015, 2015*.

V. Rudra Murthy, Mitesh M. Khapra, and Pushpak Bhattacharyya. 2019. Improving NER Tagging Performance in Low-Resource Languages via Multilingual Learning. *ACM Trans. Asian & Low-Resource Lang. Inf. Process.*, 18(2):9:1–9:20.

Dipak Narayan, Debasri Chakrabarti, Prabhakar Pande, and Pushpak Bhattacharyya. 2002. An Experience in Building the Indo WordNet-a WordNet for Hindi. In *First International Conference on Global WordNet, Mysore, India*.

Girish Palshikar, Sachin Pawar, Sangameshwar Patil, Swapnil Hingmire, Nitin Ramrakhiyani, Harsimran Bedi, Pushpak Bhattacharyya, and Vasudeva Varma. 2019. Extraction of Message Sequence Charts from Narrative History Text. In *Proceedings of the First Workshop on Narrative Understanding*, pages 28–36, Minneapolis, Minnesota. Association for Computational Linguistics.

Ritesh Panjwani, Diptesh Kanojia, and Pushpak Bhattacharyya. 2018. pyiwn: A Python-based API to access Indian Language WordNets. In *Proceedings of the Global WordNet Conference*, volume 2018.

Nitin Ramrakhiyani, Swapnil Hingmire, Sachin Pawar, Sangameshwar Patil, Girish K. Palshikar, Pushpak Bhattacharyya, and Vasudeva Varma. 2018. Resolving Actor Coreferences in Hindi Narrative Text. In *Proceedings of the 15th International Conference on Natural Language Processing, ICON 2018, Patiala, India, December 15-18, 2018*.

Pattabhi RK Rao and Sobha Lalitha Devi. 2017. EventXtract-IL: Event Extraction from Social Media Text in Indian Languages@ FIRE 2017-An Overview. In *FIRE (Working Notes)*, pages 130–135.

Pattabhi RK Rao and Sobha Lalitha Devi. 2018. EventXtract-IL: Event Extraction from Newswires and Social Media Text in Indian Languages@ FIRE 2018-An Overview. In *FIRE (Working Notes)*, pages 282–290.

Ekkart Rudolph, Peter Graubmann, and Jens Grabowski. 1996. Tutorial on Message Sequence Charts. *Computer Networks and ISDN Systems*, 28(12):1629–1641.

Roger C. Schank and Gary Saul Morson. 1995. *Tell Me a Story: Narrative and Intelligence*. Northwestern University Press.

Ashwini Vaidya. 2015. *Hindi Complex Predicates: Linguistic and Computational Approaches*. Ph.D. thesis, University of Colorado at Boulder.

Ashwini Vaidya, Sumeet Agarwal, and Martha Palmer. 2016. Linguistic Features for Hindi Light Verb Construction Identification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1320–1329.

Josep Valls Vargas. 2017. *Narrative Information Extraction with Non-Linear Natural Language Processing Pipelines*. Ph.D. thesis, Drexel University, Drexel University.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A Model-theoretic Coreference Scoring Scheme. In *Proceedings of the 6th conference on Message understanding*, pages 45–52. Association for Computational Linguistics.