

Event Centric Entity Linking for Hindi News Articles: A Knowledge Graph Based Approach

Pranav Goel, Suhan Prabhu, Alok Debnath and Manish Shrivastava

International Institute of Information Technology

Hyderabad, Telangana, India

{pranav.goel, suhan.prabhuk, alok.debnath }@research.iiit.ac.in
m.shrivastava@iiit.ac.in

Abstract

We describe the development of a knowledge graph from an event annotated corpus by presenting a pipeline that identifies and extracts the relations between entities and events from Hindi news articles. Due to the semantic implications of argument identification for events in Hindi, we use a combined syntactic argument and semantic role identification methodology. To the best of our knowledge, no other architecture exists for this purpose. The extracted combined role information is incorporated in a knowledge graph that can be queried via subgraph extraction for basic questions. The architectures presented in this paper can be used for participant extraction and event-entity linking in most Indo-Aryan languages, due to similar syntactic and semantic properties of event arguments.

1 Introduction

Events are defined as situations that happen or occur (Saurí et al., 2006). Events therefore involve participating entities, sometimes referred to as event arguments (Ji and Grishman, 2008). The extraction of role information of entities participating in events is a fast-evolving area of research in information retrieval as well as subfields of NLP such as question answering and summarization (Lin and Liang, 2008). This paper handles the challenge of participant detection and labeling in Hindi, using syntactic measures such as dependency parsing and semantic measures such as verb frame comparisons and semantic role labeling. Using the entities extracted from the text and their relation to the event, a knowledge graph is generated, which can then be queried for basic questions.

In Hindi NLP, the representation, identification and extraction of events is a fairly new concept. Event extraction from twitter data (Kuila and

Sarkar, 2017) and in news data (Ramrakhiyani and Majumder, 2013; Goud et al., 2019) are still developing areas of research. However, extensive work has been done on argument structure for Hindi verbs, therefore the syntactic analysis of verbal events has been a topic of sufficient inquiry (Butt, 2010). On the other hand, nominal events, while not studied under that paradigm, have been referenced in entity linking in NER research (Athavale et al., 2016).

This paper, given the definition of events in Hindi (Goud et al., 2019), identifies the arguments of these events. We employ a syntactico-semantic approach of entity identification by using dependency parsing to determine the syntactic roles of the arguments and their dependency length from the event mention (Gulordava et al., 2015), and a semantic role labeler which is used to determine the semantic case or functions of the participating entities (Carreras and Màrquez, 2005). For verbal events, verb frame data has also been used for verifying the arguments. This information is constructed as a knowledge graph, a query graph (Yih et al., 2015) of which can then be used for question answering.

2 Related Work

Entity or participant extraction is a vital sub-domain of event detection and related information extraction tasks. The ACE project (Dodington et al., 2004) and many of the relevant event extraction tasks that followed it had entity detection and tracking as one of the main components for event detection and extraction systems (Ahn, 2006). ACE also provided twenty-four different types of relations between entities. Hong et al. (2011) establishes a mechanism of using entity links in order to more accurately detect event mentions, by associating some entities as event par-

ticipants or arguments. Joint extraction of event and entity mentions has been attempted (Yang and Mitchell, 2016) by learning intra-event structures and possible forms of entity relations to events.

Named entity recognition has been another broader form of approach to entity identification and linking. Entity mention detection and tracking its use in the corpus (Xu et al., 2017) is considered the most fundamental method in this approach. Yamada et al. (2015) approaches the problem of named entity recognition from the perspective of entity linking. Hybrid joint approaches to participant extraction and linking (Plu et al., 2015) have been treated as an extension of this problem, and the OKE 2017 task (Plu et al., 2017) performed participant extraction and linking for ontology enrichment. Florian et al. (2004) and Lin et al. (2016) perform cross lingual entity linking over an enriched knowledge base, one of the languages being Hindi. These approaches are important for understanding and disambiguating the links between nominal events and their participants.

Argument analysis for verbs in Hindi has been a well-researched topic, as mentioned above. Palmer et al. (2009) studies the computational properties of verbal predicates from a dependency annotation perspective, while Vaidya et al. (2016) and Vaidya et al. (2019) focuses on the syntactic argument structure of light verbs in Hindi. Light verbs are one of the syntactic constructions observed in representation of eventive verbs. Compound verb detection (Chakrabarti et al., 2008), complex predicate detection (Mukerjee et al., 2006) and argument identification in complex predicates (Montaut, 2016) can be modeled together in syntactic argument detection for verbal events. The study of noun incorporation in verb complexes (Dayal, 2015) provide a semantic perspective of argument structure and event participation. Syntactically, two major concerns of verb argument analysis are verb phrase ellipsis and complex predicate analysis (Manetta, 2018b,a).

Knowledge graphs are extensively used in semantic information retrieval and has numerous other applications cross language document retrieval (Franco-Salvador et al., 2014), cross lingual plagiarism detection Franco-Salvador et al. (2016), question answering (Indurthi et al., 2017) and summarization (Zheng et al., 2016).

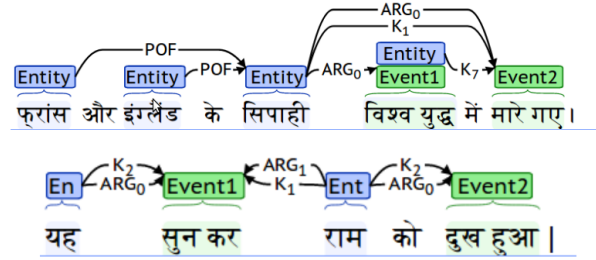


Figure 1: Example of a tagged pair of sentences. The event indexes are intra-sentence.

Overall statistics	
Number of Articles	810
Total Number of Sentences	13949
Total Number of Events	20190
Nominal Events	1841
Verbal Events	18349
Total Number of Entities	41847
Average per sentence	
Length (Words)	18
Number of Entities	3
Number of Events	1.48
Number of Entities per Event	2.08
Most Common Relation	
Entity - Nominal Event	(ARG0)
Entity - Verbal Event	(K1, ARG0)
Inter-Annotator Statistics	
Participant Identification	0.86
Syntactic Role Identification	0.89
Semantic Role Identification	0.79
Coreferent Mention Identification	0.91

Table 1: Dataset and Annotation Statistics

3 Dataset and Annotation Specifications

We use a gold-standard corpus of 810 news articles of Goud et al. (2019), and annotate it for entities and their relations with the events. The entity-event relations are annotated based on a syntactic as well as a semantic role. The syntactic role is simply a dependency label (Tandon et al., 2016), while the semantic labels are provided according to Hindi and Urdu PropBank labels (Bhatt et al., 2009).

The annotated sentence shown in Figure 1 is:

<i>frans</i>	<i>aur</i>	<i>england</i>	<i>ke</i>	<i>sipahi</i>
France	and	England	of	soldiers
<i>vishva</i>	<i>yudha</i>	<i>mein</i>	<i>maare</i>	<i>gaye.</i>
World	War	in	killed	got.
<i>ye</i>	<i>sun</i>	<i>kar</i>	<i>ram</i>	<i>ko</i>
This	listen	to	Ram	(acc)
<i>bohot</i>	<i>dukh</i>	<i>hua</i>		
much	sad	happen		

Table 1 presents some of the basic statistics of the annotated data. The data has been annotated by four annotators who are proficient in Hindi and are students of linguistics using the BRAT annotation tool (Stenetorp et al., 2012) for annotating and providing labels. The inter-annotator agreement was measured by a strict match Cohen’s Kappa Score (Cohen, 1960).

The dataset is then annotated further for ease of semantic role extraction. All event mentions are indexed, and if two event mentions are coreferent, they are given the same index. In case of entities, only entities with coreferent mentions are indexed. Coreferent entity mentions are given the same index. For this task, inter-annotator agreement was calculated on four different measures, identifying the participant, correct syntactic role, correct semantic role and correct coreferent mention identification. For the purpose of coreference, the entities and events are treated the same.

4 Identifying Entity Participation in Events

In this section, we look at the pipeline for the extraction of entities as arguments of events. We define here a nominal event as an event which has the event nugget (the core of the event) is a noun. Similarly, a verbal event is an event which has the event nugget which is a verb.

As discussed before, discerning entity participation in events is a syntacto-semantic problem, and therefore our solution (refer to Figure 2) has both syntactic and semantic components. Note that a IOB (inside-outside-begin) tagged event mentioned corpus is the input to the pipeline. The outputs from each module and the final pipeline are formatted to be used in the form of a knowledge graph, which is detailed in Section 5.

4.1 Syntactic Participation Detection

The syntactic components are essential preprocessing tasks such as POS tagging and dependency parsing. The dependency parse provides the

syntactic role information. Particularly for verbal events, the parse also provides the distance from the event mention, which is essential in order to determine participation in sentences with multiple events. This participation is then verified using verb frame data.¹

The procedure for syntactic participation detection is as follows:

1. A Hindi POS tagger (Shrivastava and Bhattacharyya, 2008) is used to identify the part of speech of all the lexical items in the sentence. Since most event arguments are nominal or pronominal, the relevant words are extracted. The POS tagged text is then provided as input to the next phase.
2. The text is then parsed using a dependency parser (Palmer et al., 2009). The dependency labels from the root are considered most important, since the nouns and pronouns directly associated with the verb are most likely to be the arguments of the event.
3. The *karaka* and *sambandh* edge labels, which are provided by the dependency parser, are extracted. The *karaka* edge labels provide the case of the noun and its role with respect to the verb, while the *sambandh* edge label mark the genitive relation between two nouns. If either one of the nouns is eventive, the relation given to it is the relevant *karaka* relation. If both the nouns are entities, then they are not linked. The relation between two entity nouns by a *sambandh* (genitive) case marker is not marked in the graph directly. Instead, genitive chains are constructed after extraction of the entity, using the dependency tree.

The dependency parser provides syntactic role information and the distance of the extracted words from the verbs in the sentence. For sentences with relative or subordinate clauses, as well as multiple events, this feature is used to determine which event is linked to which entity. The genitive *sambandh* relations are retained irrespective of the eventiveness of the nouns for the purpose of identifying the primary participant in an event in case of a long genitive chain.

¹While we define entities by participation similar to ACE (Doddington et al., 2004), the definition of event (Goud et al., 2019) allows for multiple events in a single sentence.

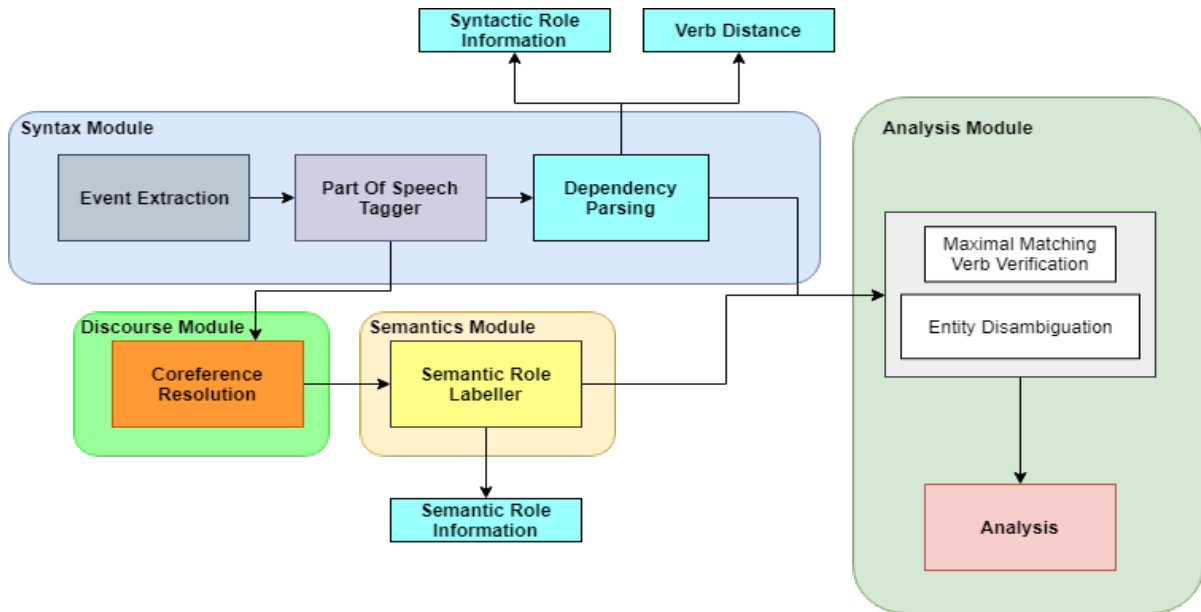


Figure 2: Pipeline for Participant Extraction

4.2 Semantic and Discourse Relation Extraction

The semantic role of the arguments to an event are extracted by the semantic role labeler (SRL) for Indian languages (Anwar and Sharma, 2016). The SRL uses POS tagged text as an input and provides the semantic role of the nouns and adverbs in the sentence. For the purpose of participant extraction, the adverbs are ignored.

However, before the semantic role extraction can be done, event coreference, entity coreference and anaphora resolution are performed, in order to determine the possible overlap of event mentions (multiple event mentions for the same event) (Chen et al., 2009).

- **Event coreference** is taken care of by indexing the event. All event mentions in the annotated input are indexed by a numerical subscript. Coreferent events have similar event triggers and overlapping argument structures (Lu et al., 2016), which are crucial features in the annotation of these events. The indices of coreferent event mentions are the same, which indicates that they share their arguments.
- **Entity coreference** is taken care of determining the role that the entity performs in the event. This is one of the primary entity based features used for entity coreference (Clark and Manning, 2015). In the corpus, the en-

tities are partially indexed, that is that only coreferent entities are indexed.

Both anaphora and event coreference are done automatically using a combination of role extraction and verb relations as mentioned above, as well as using pretrained models (Devi et al., 2014) and manual editing of the output.

After this, if a noun also happens to be an event, the dependency relation between it and a verbal event in the sentence (if any) is retained, while the semantic relation is removed. Event-event relations are beyond the scope of this paper, and for the sake of simplicity, it is assumed that events can not be arguments to other events. Retaining the dependency information, however, as it is a feature used in entity disambiguation if an entity happens to participate in a nominal and a verbal event. As with the dependency parse, the semantic relations between two nouns is retained regardless of their eventiveness, as the semantic relation acts as a verification for the detected primary participant.

4.3 Role Analysis and Verification

In order to accurately determine the roles assigned by the two modules above, our pipeline is equipped with an analysis module. In line with the Paninian tradition, we use the notion of *yogyata* (capability) (Kulkarni et al., 2010) to verify whether an event can take the types and roles of the arguments that have been assigned to it. The output of this system are then analyzed as tem-

plates of entity-event relations, which are used to create the knowledge graph.

Verbal events are analyzed using verb frame data (Soni et al., 2013). The verb frame data provides the possible *karaka* relations which can be used to determine the mandatory and optional syntactic expectancy of the verb in different senses. A maximal matching algorithm (Algorithm 1) is used across all senses, and the sense with *all* mandatory and the maximum number of optional *karaka* arguments is chosen as the sense of that verb.

Algorithm 1 Maximal matching Verb Verification

```

1: procedure MAXMATCHVERB
2:    $VFD \leftarrow$  Verb Frame Data
3:    $V \leftarrow$  Verbal Event
4:    $part \leftarrow$  list [(Parent, Participant, Role)]
5:    $max\_all \leftarrow 0$ 
6:    $max \leftarrow -1$ 
7:   for  $verb$  in  $VFD$  do
8:     for  $sense$  in  $verb$  do
9:       if ( $V = verb$ ) and ( $part[2] =$ 
10:         $sense$ ) then
11:          $max = max + 1$ 
12:         if ( $max\_all < max$ ) then
13:            $max\_all \leftarrow max$ 
14:   return  $max\_all$ 

```

Algorithm 2 Entity Disambiguation

```

1: procedure ENTITYDISAMBIGUATION
2:    $N \leftarrow$  NominalEvent
3:    $E \leftarrow$  EntityList
4:   if  $N$  in  $E$  then
5:     Remove  $N$  from  $E$ 
6:    $V \leftarrow$  Closest verbal event from  $N$   $\triangleright$ 
7:     Word distance or tree distance
8:   Add  $ARG_0, ARG_1$  of  $V$  to  $E$ 
9:    $VEList \leftarrow$  list of all verbal events
10:  for  $VE \in VEList$  do
11:    if  $ARG_{2LOC} \in VE$  then
12:      Add  $ARG_{2LOC}$  of  $VE$  to  $E$ 
13:    if  $ARG_{2GOL} \in VE$  then
14:      Add  $ARG_{2GOL}$  of  $VE$  to  $E$ 
15:    if  $ARG_{SOU} \in VE$  then
16:      Add  $ARG_{SOU}$  of  $VE$  to  $E$ 
17:    if  $ARG_{TMP}$  exists then
18:      if  $ARG_{TMP} \notin arg(VE)$  for  $VE \in$ 
19:       $VEList$  then
20:        Add  $ARG_{TMP}$  to  $E$ 

```

Nominal entity participant identification follows two steps, jointly referred to as entity disambiguation. First, we use a naive coreference resolution using a feature set similar to Dakwale et al. (2013)’s rule based implementation, for entities and events. The syntactic roles of significance are *sambandh* relations. Some of the design choices in Lee et al. (2012), including features such as number of coreferent arguments and argument roles are crucial to determining participation, as shown in Algorithm 2.

Finally, we analyze and resolve co-participation ambiguities. For sentences with multiple events, it is necessary to verify whether all the entities necessarily participate in, or are modified by, the attributed events. In verbal events, maximal matching is done on the entities syntactically closest to it, which performs well in default word order. For entities linked to both nominal and verbal events, semantic role information is considered. Nominal events characteristically only take agentive, thematic and locative arguments over the verbal predicate (Gerber and Chai, 2012), while only those temporal arguments are taken which are not attributed to the verbal event.

After completing this analysis, the output of the pipeline is condensed and reformatted as inputs to a knowledge graph.

5 Entity-Event Knowledge Graph

Knowledge graphs have been widely used in information retrieval, since their adoption in popular search engines. However, knowledge graphs can be constructed for document wide, corpus wide or domain wide extraction of information as well. In this section, we show the development of an event-centric entity linked knowledge graph.

(Rospocher et al., 2016) defines an event centric knowledge graph as a knowledge graph in which all information is related to events through which the knowledge in the graph obtains a temporal dimension. Knowledge graphs are useful for the representation of semantic information in the edge labels or in the attributes of the nodes itself. Document wide knowledge graphs can be queried by limiting the search space based on the query. This method of creating a query graph allows for an inference chain for the related nodes (Yih et al., 2015).

Question word	Gloss	Category	Role
<i>kis</i> + case marker ²	who	Entity	-
<i>kaun</i>	who	Entity	-
<i>kahAn</i>	where	Entity	Location or Source
<i>kab</i>	when	Entity	(Time)
<i>kyun</i>	why	Event or Entity	(Goal)

Table 2: Question Words and Answer Types

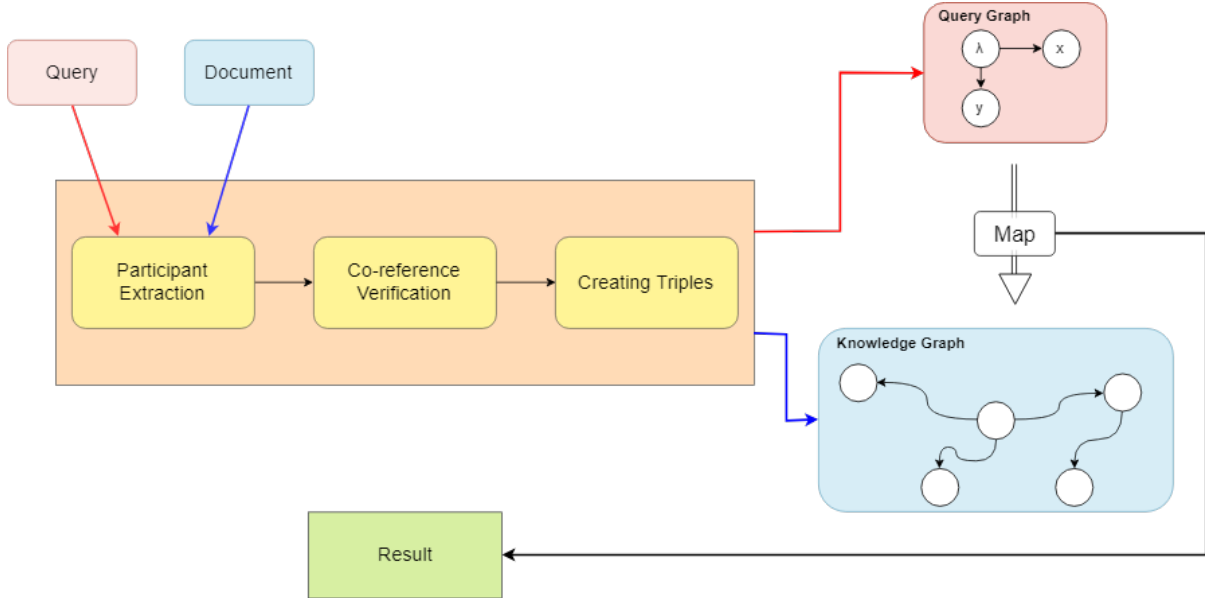


Figure 3: Pipeline for Knowledge Graph and Query Graph Creation. "Participant Extraction" refers to the Pipeline in Fig. 1

5.1 Developing the Knowledge Graph

In order to develop a knowledge graph, we must determine the relevant nodes and edges. We choose to consider events and entities as nodes, and the relations between them as the edges. The relations between them, as mentioned before, are both syntactic and semantic. We show the development of the knowledge graph and handling queries in Figure 3

Creating triples As with most knowledge graph based representations, the first step is to extract the necessary triples that constitute the graph. The data after being passed through the entity detection and linking pipeline, has to be reformatted into $(e, (n, m), v_i)$ triples, where $e \in E$, the set of all entities, $(n, m) \in (N \cup \{\phi\}, M)$, where N is the set of all syntactic roles and M is the set of all semantic roles, and $v_i \in V$, the set of all events in the document, indexed. If there is no syntactic role of the entity in an event, as is common with nominal events, the syntactic role given to it is ϕ .

We also construct specific *genitive triples*, de-

defined as (e_i, n, e_j) where $e_i, e_j \in E$, the set of all entities and n is always given a *POF* relation. These links are useful when constructing entity links and chains. The genitive triples are not used directly in the knowledge graph. Instead, the constructed genitive triples (since they are directly extracted from the dependency graph) are used for generating an answer for a query. These are maintained primarily for efficiency in generating an answer for the query.

Handling Event Coreference After triple creation, event coreference has to be handled. Coreference is handled in semantic role extraction. Events are indexed by the occurrence of their first mention in the text. A relation has to be created between the entities of events with the same index. Note, however, that because of the temporal nature of events, entities that are linked to a later mention of a coreferent event are not linked to the first. In our approach, the first mention of an event is considered its *primary mention*, for the purpose of creating the knowledge graph. All

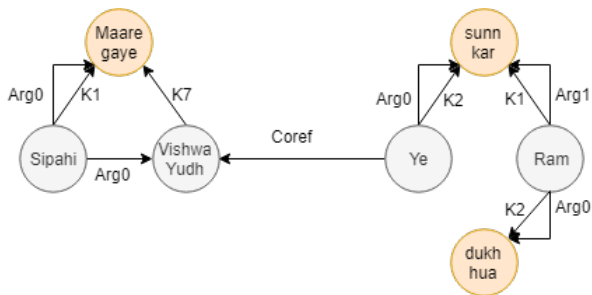


Figure 4: Knowledge graph representation of fig. 1

other mentions are *secondary mentions*, which are ordered through the document in their order of occurrence. The entities participating in the primary event mention are considered participants to all the secondary mentions, while the arguments of the first secondary mention are arguments only to itself and the remaining secondary mentions and so on. Therefore, for each new event mention of the same event, new triples are made which account for the participation of that event from all previous event mentions.

Handling Entity Coreference In the case of entity coreference and anaphora, entities with multiple mentions are already indexed, and therefore, all the entity mentions are considered the same entity, and if an entity mention participates in an event, all other entity mentions participate in it as well. Therefore using the index values of the entities, a coreference chain can be formed that defers all entity mentions to the *primary entity mention*, which is the first mention of an entity. This choice also makes query graph formation easier. Therefore, all triples where the entity is indexed are replaced with the primary mention of the entity.

Figure 4 shows the knowledge graph of a snippet of the sentences from figure 1. Events and entities are both nodes, as mentioned; we use colors to distinguish between them.

5.2 Querying the Knowledge Graph

Once the knowledge graph has been created, it can be used for other downstream tasks and applications. One of the major applications is question-answering. Recent approaches to open domain question-answering systems over graph databases like Freebase (Bollacker et al., 2008) follow a semantic parsing approach (Yao and Van Durme, 2014). Our approach for querying the entity-event knowledge graph is similar to Yih et al. (2015)’s approach. We generate a query graph of the ques-

tion and perform predicate matching over the λ -expression corresponding to the query graph after exhausting all possible inference chains.

We are first tasked with the annotation of events and entities in the question. Event annotation is done by the methods described in (Goud et al., 2019), and are not discussed in the scope of this paper. But given an event annotated event sentence, we first identify the entities and the *question entity*, which is the interrogative pronoun. The basic pipeline for entity recognition in the document is also followed for the question. In the analysis phase, the question pronoun is marked. We map the question pronoun to the type of response expected, that is, either an entity or an event. Using this information, the query graph is created, from which a λ -expression is extracted.

Carrying the example from section 3, a factoid question based on the sentences (sentences in 1, graph in 4) could be:

yudh mein kaun maare gaye?
War in who killed got?

As mentioned, we first parse and analyze the question, as has been done before in section 4. The dependency parse provides us with which word is the question word. We also use a specific morphanalysis module to extract syntactic role (*karaka*) information.

From this, we construct the λ -expression $\lambda x.\exists y : \text{entity}(x, y) \wedge \text{Arg0}(y, \text{maare gaye}) \wedge \text{k1}(y, \text{maare gaye}) \wedge \text{Arg0}(y, \text{yudh})$. The first of the relations (*entity*), can be determined based on the question word’s role in the sentence. For the purpose of factoid questions on our dataset, only the question *kyun* (why) is considered to have an answer which is tagged event. Table 2 is the simple mapping from question to query. In the cases where a question can have multiple types of answers, the largest number of overlapping words is considered the disambiguating heuristic. Question words such as *kaise* (how) and *kyA* (what) are not accounted for, as not all formats of the question are factoid in nature. Therefore, using the lambda-expression, we can construct a query graph, which can easily be mapped onto the knowledge graph, and the y is the answer to the query, while $\lambda(x)$ ascertains whether the answer is of the correct type (event or entity).

Entity Detected	Overlap	Label
Overall	86.4%	84.1%
Nominal Events	64.1%	71.7%
Verbal Events	93.7%	89.4%

Table 3: Average Accuracy for Participant Extraction

6 Analysis and Results

In this section, we look into the two pipelines which have been developed for constructing a basic knowledge graph from an event annotated corpus, and the type of queries it can handle. We provide both a qualitative and quantitative analysis of the results of the pipelines. We also provide a thorough analysis of errors.

6.1 Participant Extraction Pipeline

The participant extraction pipeline (figure 2) has multiple interdependent components, such as the event annotated corpus, the POS tagger and dependency parser, the coreference resolution module and the semantic role labeler. Based on the annotated data, we find that the pipeline accurately detects the presence of 86.4% of the participants of the events for each event on an average. Table 3 shows the percentage of average complete overlap and the accuracy of label detection. Note that only complete overlap of the entity span is considered as the output and the label is considered accurate if all the roles have been correctly identified.

The relative drop in accuracy for nominal events is due two primary reasons, first that there are no syntactic features for the detection of participants in nominal events and secondly coreference of nominal events as entities. We notice that a coreferent event mention can act as an entity, but still hold eventive characteristics, which has not been handled in our pipeline. Furthermore, due to case marker overloading (Bharati et al., 2002) in Hindi, the accurate detection of labels is affected.

6.2 Knowledge Graph and Queries

In the creation of the knowledge and query graphs, illustrated in figure 3, we see that the errors of the participant extraction pipeline mentioned above will propagate forward, causing the knowledge graph to be an ill-representation of the document. As mentioned above, the characteristic error arose from coreference mishandling, and therefore, the coreference validation module accounts for the assigning the eventive nature of the coreferent event

mentions which act as entities.

We qualitatively analyze the knowledge graph and the pipeline by using simple queries in order to verify the creation of the graph and the associated nodes and edges. The queries, as shown in Figure 3, also pass through the same pipeline, and we verify the knowledge graph based on the accuracy of the response to the query. Since the λ expressions are constructed based on simple rules based on the nature of Hindi question words, we could only qualitatively analyze the graph on simple queries with single query results.

Queries of the form 'kis' + case marker or *kaun* provide a valid response to the query. However, for sentences with multiple events, queries provide incorrect results in some cases. This is partly because of entity sharing, which is that an entity is associated to multiple events if they are subevents. Since the relations between events have not been handled yet, and is beyond the scope of this paper.

7 Conclusion

In this paper, we attempted to determine a method of identifying the participants of each event in an event-annotated corpus, given the syntactic and semantic role of each noun and verb in a sentence. We used a distinct pipeline of interacting tools which provided various levels of syntactic and semantic information, which were then combined and analyzed. We have presented the two major algorithms; one for identifying the sense of the verb being used (based on the available mandatory arguments and the maximum match of optional arguments), and the other for determining the participants of a nominal event. We have also presented the development of a queryable knowledge graph on the basis of the events and entities extracted, that use the role information as edge labels. With this work, we hope to develop a more robust representation of events and entities, which can be enriched with developments in event classification and event relation extraction in Hindi. Most importantly, the pipeline and algorithms developed in this paper are language agnostic, which we hope will spur research into developing information rich representations of event and participation information in other languages as well.

References

David Ahn. 2006. The stages of event extraction. In *Proceedings of the Workshop on Annotating and*

- Reasoning about Time and Events*, pages 1–8.
- Maaz Anwar and Dipti Sharma. 2016. Towards building semantic role labeler for indian languages. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).
- Vinayak Athavale, Shreenivas Bharadwaj, Monik Pamecha, Ameya Prabhu, and Manish Shrivastava. 2016. Towards deep learning in hindi ner: An approach to tackle the labelled data scarcity. In *13th International Conference on Natural Language Processing*, page 154.
- Akshar Bharati, Rajeev Sangal, Vineet Chaitanya, Amba Kulkarni, Dipti Misra Sharma, and KV Ramakrishnamacharyulu. 2002. Anncorra: building tree-banks in indian languages. In *COLING-02: The 3rd Workshop on Asian Language Resources and International Standardization*.
- Rajesh Bhatt, Bhuvana Narasimhan, Martha Palmer, Owen Rambow, Dipti Sharma, and Fei Xia. 2009. A multi-representational and multi-layered treebank for hindi/urdu. In *Proceedings of the Third Linguistic Annotation Workshop (LAW III)*, pages 186–189.
- Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. AcM.
- Miriam Butt. 2010. The light verb jungle: Still hacking away. *Complex predicates in cross-linguistic perspective*, pages 48–78.
- Xavier Carreras and Lluís Màrquez. 2005. Introduction to the conll-2005 shared task: Semantic role labeling. In *Proceedings of the ninth conference on computational natural language learning (CoNLL-2005)*, pages 152–164.
- Debasri Chakrabarti, Hemang Mandalia, Ritwik Priya, Vaijayanthi Sarma, and Pushpak Bhattacharyya. 2008. Hindi compound verbs and their automatic extraction. *Coling 2008: Companion volume: Posters*, pages 27–30.
- Zheng Chen, Heng Ji, and Robert Haralick. 2009. A pairwise event coreference model, feature impact and evaluation for event coreference resolution. In *Proceedings of the workshop on events in emerging text types*, pages 17–22. Association for Computational Linguistics.
- Kevin Clark and Christopher D Manning. 2015. Entity-centric coreference resolution with model stacking. 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)*, volume 1, pages 1405–1415.
- Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.
- Praveen Dakwale, Vandan Mujadia, and Dipti M Sharma. 2013. A hybrid approach for anaphora resolution in hindi. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 977–981.
- Veneeta Dayal. 2015. Incorporation: Morpho-syntactic vs. semantic considerations. In *The syntax and semantics of pseudo-incorporation*, pages 47–87. BRILL.
- Sobha Lalitha Devi, Vijay Sundar Ram, and Pat-tabhi 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.
- George R Doddington, Alexis Mitchell, Mark A Przybocki, Lance A Ramshaw, Stephanie M Strassel, and Ralph M Weischedel. 2004. The automatic content extraction (ace) program-tasks, data, and evaluation. In *LREC*, volume 2, page 1.
- Radu Florian, Hany Hassan, Abraham Ittycheriah, Hongyan Jing, Nanda Kambhatla, Xiaoqiang Luo, H Nicolov, and Salim Roukos. 2004. A statistical model for multilingual entity detection and tracking. Technical report, IBM THOMAS J WATSON RESEARCH CENTER YORKTOWN HEIGHTS NY.
- Marc Franco-Salvador, Paolo Rosso, and Manuel Montes-y Gómez. 2016. A systematic study of knowledge graph analysis for cross-language plagiarism detection. *Information Processing & Management*, 52(4):550–570.
- Marc Franco-Salvador, Paolo Rosso, and Roberto Navigli. 2014. A knowledge-based representation for cross-language document retrieval and categorization. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 414–423.
- Matthew Gerber and Joyce Y Chai. 2012. Semantic role labeling of implicit arguments for nominal predicates. *Computational Linguistics*, 38(4):755–798.
- Jaipal 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.
- Kristina Gulordava, Paola Merlo, and Benoit Crabbé. 2015. Dependency length minimisation effects in short spans: a large-scale analysis of adjective placement in complex noun phrases. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, volume 2, pages 477–482.

- Yu Hong, Jianfeng Zhang, Bin Ma, Jianmin Yao, Guodong Zhou, and Qiaoming Zhu. 2011. Using cross-entity inference to improve event extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1127–1136. Association for Computational Linguistics.
- Sathish Reddy Indurthi, Dinesh Raghu, Mitesh M Khapra, and Sachindra Joshi. 2017. Generating natural language question-answer pairs from a knowledge graph using a rnn based question generation model. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 376–385.
- Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. *Proceedings of ACL-08: HLT*, pages 254–262.
- Alapan Kuila and Sudeshna Sarkar. 2017. An event extraction system via neural networks. In *FIRE (Working Notes)*, pages 136–139.
- Amba Kulkarni, Sheetal Pokar, and Devanand Shukl. 2010. Designing a constraint based parser for sanskrit. In *International Sanskrit Computational Linguistics Symposium*, pages 70–90. Springer.
- Heeyoung Lee, Marta Recasens, Angel Chang, Mihai Surdeanu, and Dan Jurafsky. 2012. Joint entity and event coreference resolution across documents. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 489–500. Association for Computational Linguistics.
- Fu-ren Lin and Chia-Hao Liang. 2008. Storyline-based summarization for news topic retrospection. *Decision Support Systems*, 45(3):473–490.
- Ying Lin, Xiaoman Pan, Aliya Deri, Heng Ji, and Kevin Knight. 2016. Leveraging entity linking and related language projection to improve name transliteration. In *Proceedings of the Sixth Named Entity Workshop*, pages 1–10.
- Jing Lu, Deepak Venugopal, Vibhav Gogate, and Vincent Ng. 2016. Joint inference for event coreference resolution. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3264–3275.
- Emily Manetta. 2018a. The structure of complex predicates in hindi-urdu: evidence from verb-phrase ellipsis. *Trends in Hindi Linguistics*, 325:47.
- Emily Manetta. 2018b. Verb-phrase ellipsis and complex predicates in hindi-urdu. *Natural Language & Linguistic Theory*, pages 1–39.
- Annie Montaut. 2016. Noun-verb complex predicates in hindi and the rise of non-canonical subjects. In *Approaches to Complex Predicates*, pages 142–174. Brill.
- Amitabha Mukerjee, Ankit Soni, and Achla M Raina. 2006. Detecting complex predicates in hindi using pos projection across parallel corpora. In *Proceedings of the Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties*, pages 28–35. Association for Computational Linguistics.
- Martha Palmer, Rajesh Bhatt, Bhuvana Narasimhan, Owen Rambow, Dipti Misra Sharma, and Fei Xia. 2009. Hindi syntax: Annotating dependency, lexical predicate-argument structure, and phrase structure. In *The 7th International Conference on Natural Language Processing*, pages 14–17.
- Julien Plu, Giuseppe Rizzo, and Raphaël Troncy. 2015. A hybrid approach for entity recognition and linking. In *Semantic Web Evaluation Challenge*, pages 28–39. Springer.
- Julien Plu, Raphaël Troncy, and Giuseppe Rizzo. 2017. Adel@ oke 2017: a generic method for indexing knowledge bases for entity linking. In *Semantic Web Evaluation Challenge*, pages 49–55. Springer.
- Nitin Ramrakhiyani and Prasenjit Majumder. 2013. Temporal expression recognition in hindi. In *Mining Intelligence and Knowledge Exploration*, pages 740–750. Springer.
- Marco Rospocher, Marieke van Erp, Piek Vossen, Antske Fokkens, Itziar Aldabe, German Rigau, Aitor Soroa, Thomas Ploeger, and Tessel Bogaard. 2016. Building event-centric knowledge graphs from news. *Journal of Web Semantics*, 37:132–151.
- Roser Saurí, Jessica Littman, Bob Knippen, Robert Gaizauskas, Andrea Setzer, and James Pustejovsky. 2006. Timeml annotation guidelines. *Version*, 1(1):31.
- Manish Shrivastava and Pushpak Bhattacharyya. 2008. Hindi pos tagger using naive stemming: Harnessing morphological information without extensive linguistic knowledge. In *International Conference on NLP (ICON08), Pune, India*.
- Ankush Soni, Sambhav Jain, and Dipti Misra Sharma. 2013. Exploring verb frames for sentence simplification in hindi. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1082–1086.
- Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun’ichi Tsujii. 2012. Brat: a web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107. Association for Computational Linguistics.
- Juhi Tandon, Himani Chaudhry, Riyaz Ahmad Bhat, and Dipti Sharma. 2016. Conversion from paninian

- karakas to universal dependencies for hindi dependency treebank. In *Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016)*, pages 141–150.
- 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.
- Ashwini Vaidya, Owen Rambow, and Martha Palmer. 2019. Syntactic composition and selectional preferences in hindi light verb constructions. *LiLT (Linguistic Issues in Language Technology)*, 17.
- Mingbin Xu, Hui Jiang, and Sedtawut Watcharawit-tayakul. 2017. A local detection approach for named entity recognition and mention detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1237–1247.
- Ikuya Yamada, Hideaki Takeda, and Yoshiyasu Takefuji. 2015. Enhancing named entity recognition in twitter messages using entity linking. In *Proceedings of the Workshop on Noisy User-generated Text*, pages 136–140.
- Bishan Yang and Tom M Mitchell. 2016. Joint extraction of events and entities within a document context. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 289–299.
- Xuchen Yao and Benjamin Van Durme. 2014. Information extraction over structured data: Question answering with freebase. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 956–966.
- Scott Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. 2015. Semantic parsing via staged query graph generation: Question answering with knowledge base.
- Weiguo Zheng, Lei Zou, Wei Peng, Xifeng Yan, Shaoxu Song, and Dongyan Zhao. 2016. Semantic sparql similarity search over rdf knowledge graphs. *Proceedings of the VLDB Endowment*, 9(11):840–851.