International Journal of Mathematics and
Computer Science, **15**(2020), no. 1, 199–212

A Conceptual Graph Approach to the Parsing of Projective Sentences

Ashish Pradhan[1], **Archit Yajnik**[1], **Manisha Prajapati**[2]

[1]Department of Mathematics
Sikkim Manipal Institute of Technology
Majitar, Rangpo-737136
East Sikkim, India

[2]Gujarat Technological University
Ahmedabad, Gujarat, India

email: Asiispradhan@gmail.com, Archit.yajnik@gmail.com,
purvansi263@gmail.com

(Received July 15, 2019, Accepted October 4, 2019)

Abstract

This paper presents a Conceptual Graph (CG) to the parsing of the Projective sentences for the Gujarati and Nepali texts. Pananian framework is used to define the syntax and semantic structures. A purely new concept and technique of parsing of a natural language is presented. With complex and a large number of hierarchies in dependency relation among words in many Indian languages like Nepali, this new proposed formalism provides clear insights and sorts such hierarchy. The formalism is explored to parse non-projective sentences by decomposing it into projective halves.

Key words and phrases: Planar graphs, Connectivity, Trees, Hypergraphs, Directed Graphs, Ordered Bipartite Graph, NL Support, Concept Hierarchy, Relation arity, Parsing, Projectivity.
AMS (MOS) Subject Classifications: 05C10, 05C40, 05C05, 05C65, 05C20, 05C99.
ISSN 1814-0432, 2020, http://ijmcs.future-in-tech.net

1 Introduction

After the introduction of Conceptual Graph (CG) by Sowa [1, 2], the CGs have been applied to many knowledge based models and comparison techniques like, Knowledge management task, such as information retrieval and text mining, database interface [2], generation of referring expression[3], implementing semantic interpreter [4], Comparison of personal ontologies [5], ontology similarity measure [6] and so on. In this study, we propose a Conceptual Graph to the Parsing of the natural languages Nepali and Gujarati. Nepali is an Indo-Aryan language spoken by approximately 45 million speakers in Nepal, Bhutan, Myanmar and some parts of India [7] and Gujarati is spoken by more than 3 million. Both the Languages are included in the 23 official languages of India and are incorporated in the Indian constitution.
In Natural Language Processing, Parsing is a method of analyzing the grammatical structure of sentences. Data driven and grammar driven approaches have been employed over the years for parsing of a natural language. The adjective data-driven means that progress in an activity is compelled by the data, rather than by intuition or by experience. The Grammar driven approach requires a deep knowledge of the formal grammar rules, syntax and semantics of the language. The Parsing is one of the principal steps involved in Machine translation. The machine translation of a natural language without parsing the sentences using syntax and semantics of can yield wrong results. For example translating the sentences (Nepali) using Google Translate [8] shows following outputs, "उसले फर्केर हेर्यो अन्त भन्यो, तिमीलाई के भाकोछ आजकल?" meaning: “He turned back and said, what is going on with you nowadays?”. Output: “He turned back, said the end, what is your vow nowadays?”.

Hence the Parsing of a natural language with respect to the syntactic and semantic knowledge of the language is vital.

Sowa [1], the original author of the conceptual graph theory who formed the basis for the Conceptual Structure, said: “A conceptual graph has no meaning in isolation. Only through the semantic network are its concepts and relations linked to context, language, emotion, and perception.” Conceptual Graphs are a visual, logic based knowledge representation formalism. They encode ontological knowledge in a structure called support. The support consist of concept type hierarchy

and relation type hierarchy, a set of individual markers that refer to specific concept and generic marker, denoted by * which refer to an unspecified concept in the application domain. A CG is structure that depicts factual information about the background knowledge contained in its Support. This knowledge is presented in a visual manner as an ordered Bipartite Graph, whose nodes have been labeled with elements from the Support. A CG is defined as [9] $CG = (G, S, \lambda)$, where,

- S is a support,
- G is an ordered bipartite graph,
- λ is a labeling of the vertices of G with elements from the support S: $\forall\, r \in V_R$, $\lambda\,(r) \in {T_R}^{d_G(r)}$; $\forall\, c \in V_C$, $\lambda\,(c) \in T_C \times I \cup \{\,*\,\}$.

Definition 1.1. *A graph $G = (V_C, V_R, N_G)$ is called an Ordered Bipartite Graph* [9] *if V_C and V_R are finite disjoint sets, where $V = V_C \cup V_R$ is the vertices set of G, and $N_G : V_R \longrightarrow V_{C+}$ is a mapping, where V_{C+} is the set of all finite nonempty sequences over V_C. For $r \in V_R$ with $N_G(r) = c_1...c_k$, $d_G(r) = k$ is the degree of r in G and $N_G^i(r) = c_i$ is the i-neighbor of r in G. The set of (distinct) neighbors of r is denoted by $N_G(r)$. The set of edges of G is given by $E_G = \{(c, r) | c \in V_C, r \in V_R$ and $\exists i$ such that $N_G^i(r) = c\}$.*

Definition 1.2. *$S = (T_C, T_R, I, *)$ is a Support* [9] *where:*

- *T_C is a finite partially ordered set, $(T_C, \leq)$, of concept types.*
- *T_R is a finite set of relation types partitioned into k partially ordered sets $(T_R^i, \leq)_{i=1,k}$ of relation types of arity $i (1 \leq i \leq k)$, where k is the maximum arity of a relation type in T_R.*
- *I is the set of countable set of individual markers, used to refer specific concepts.*
- *$*$ is the generic marker used to refer to an unspecified concept (having, however, a specified type).*

In the following section we define a CG to the Parsing of Nepali and Gujarati Languages.

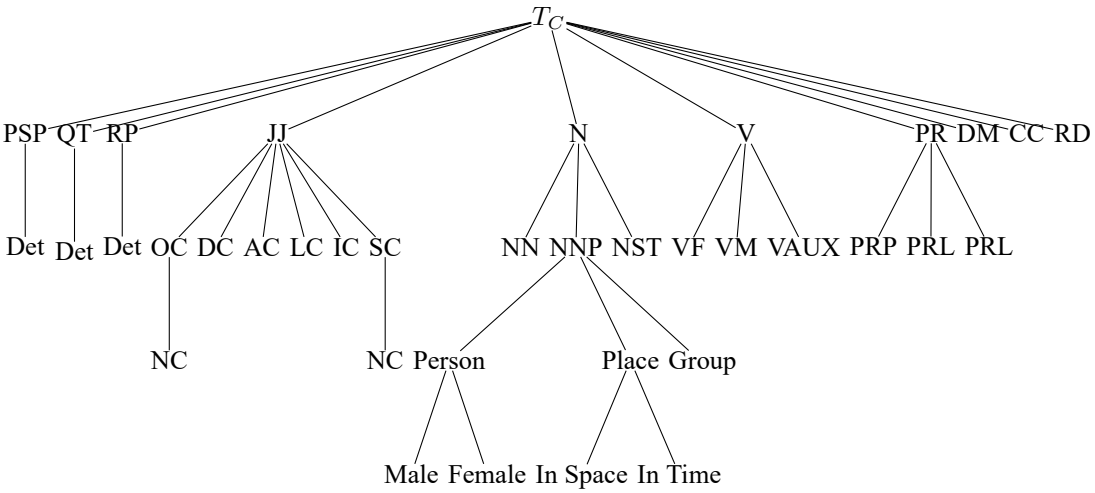

Figure 1: Concept Type Hierarchy

2 The Constraint Graph

A Bipartite Graph G is defined as $G = (V_C, V_R, E)$, where V_C, V_R are the set of nodes such that $V_C \cap V_R = \phi$ and E is the set of edges between V_C and V_R. To construct a problem bipartite graph for given sentence, we first construct a constraint graph following the Pananian Framework [10] and then we consider following the four steps:

1. For every source node s in the constraint graph, form a node s in V_C [10].
2. For every demand node d in the constraint graph and every mandatory karaka k in the karaka chart for d, from a node v in V_R (thus for every pair (d, k) there is a node in V_R [10].
3. For every demand node d in the constraint graph and every possible optional karaka k in the karaka chart for d, from a node v in V_R.
4. For every edge (d, s) labeled by karaka in the constraint graph create edge between node (d, k) in V_R to s in V_C [10].

5. If for a source node s in the constraint graph there is no mandatory karaka in the karaka chart for d, then optional karaka on the priority basis will be considered as mandatory.

Theorem 2.1. *Let $G = (V, E)$ a Bipartite Graph representing a Projective sentence, such that $V = V_1 \cup V_2$ and $V_1 \cap V_2 = \phi$, where V_1 is the set of the non-verb nodes and V_2 is the set of verb nodes and each vertex of V_1 is connected to a vertex in V_2, then G have a Planar embedding whenever $|V_2| \neq 3k$, $\forall k = 1, 2, ..., t$.*

Proof : Let us consider a projective sentence with m number of non-verb words and n number of verbs i.e. $|V_1| = m$ and $|V_2| = n$. If $n = 1$, then G is a star graph, which always have an planar embedding. let $n \neq 1$, now a theorem by Kurattowski [13] *states that a graph is a planar graph i.e. have planar embeddings if and only if it does not contain subdivisions of k_5 or $k_{3,3}$. In our case we don't have to consider for subdivisions of k_5 as we are only considering bipartite structures. So we are done if we prove that G does not contain subdivisions of $k_{3,3}$.*

Let us assume that G contain a subdivision of $k_{3,3}$, which implies that for some projective sentence, $|V_2| = 3k$ for some k which contradicts the fact that $|V_2| \neq 3k \forall k = 1, 2, ..., t$. Therefore the proof is complete.

3 Conceptual Graph for Parsing

To define a Conceptual Graph for a given sentence (Nepali or Gujarati) we define the following first:

Definition 3.1. *Ordered bipartite graph: defined as*

$$B = (V_C, V_R; E_B, l).$$

which is formed by first considering a problem bipartite graph $G = (V_C, V_R, E)$ defined in section 2 and then defining a linear ordering $\forall w_i \in V_C$ on set of edges incident to w_i. where V_R represents the verb words paired with all the possible mandatory and optional karakas and V_C is the set of all the non-verb words. An ordered bipartite graph requires that for each node in one of the classes of the bipartition, its neighbors (belonging to the other partition) to be ordered [12]. *For*

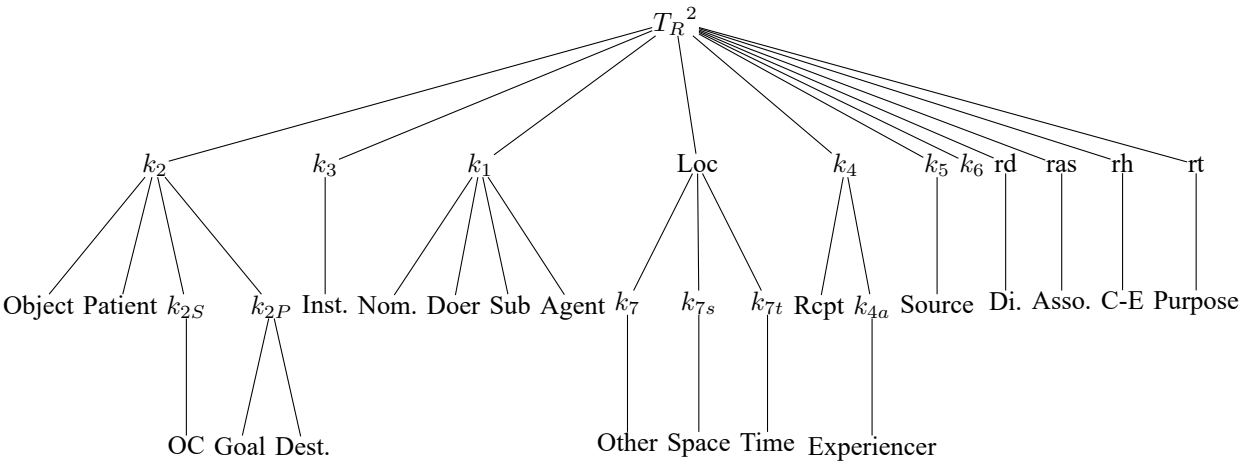

Figure 2: Relation Hierarchy

every verb-karaka combination (v_jk) *for each non-verb words* (w_i)*, we define a labeling*

$$l_i : E_B \longrightarrow \{1, 2, ..., |V_R|\},$$

on the edges of B *by*

$$l_i[v_jk, w_i]_{\forall k \in T_R, \forall w_i \in V_C} = \begin{cases} 1, & \text{if } k \text{ is mandatory } \forall j \\ 2, & \text{if } k \text{ is optional } \forall j \\ 3, & \text{otherwise} \end{cases}$$

where $i = 1, 2, ..., |V_C|$*,* $j = 1, 2, ..., |V_R|$ *and* $l_i(\{v_jk, w_i\})$ *is the index of the edge* $\{v_jk, w_i\}$ *in the above ordering of the edges incident in* B *to* V_C*. The label* $l_i \forall i \in |V_C|$*, is called the order labeling of the edges of* B*. Now we have for each* $v_jk \in V_R$*,* ${N_B}^p(v_jk)$ *denotes the* $p-th$ *neighbor of* v_jk*; i.e.,* $w_p = {N_B}^p(v_jk)$ *, iff* $\{v_jk, w_p\} \in E_B$*.*

Given a node $x_m \in V_C \cup V_R$ *,* $\overline{N_B}(x_m)$ *denotes the neighbors set of this node; i.e.,*

$$\overline{N_B}(x_m) = \{u_m \in V_C \cup V_R | \{x_m, u_m\} \in E_B\}.$$

Similarly, if $A \subseteq V_C \cup V_R$, its neighbors set is denoted as

$$\overline{N_B}(x_m) = \bigcup_{x_m \in A} \overline{N_B}(x_m) - A.$$

We further assume that for each $w_i \in V_C$ there is $v_j k \in V_R$ and $n \in \mathbb{N}$ such that $w_i = {N_B}^n(v_j k)$; i.e., B has no isolated vertices. Ordered bipartite graphs are appropriate tools to represent and visualize (directed) hypergraphs [9]. Visually, an ordered bipartite graph can be represented using boxes for vertices in V_C, ovals for vertices in V_R and integer labeled simple curves (edges) connecting boxes and ovals [9].

Definition 3.2. *$NLS = (T_C, T_R, I, *)$ is a NL Support related with the syntax and semantics of the natural language where:*

- *T_C is a finite, partially ordered set of concept types which is depicted in figure* 1*,*
- *T_R is a finite poset of relation types of arity $2({T_R}^2)$ and T_R^* (set of generic relations),*
- *I is the set of countable set of individual markers, used to refer specific concepts and*
- *$*$ is the generic marker used to refer to an unspecified concept (having, however, a specified type).*

Definition 3.3. *Finally we define λ as the labeling of all the vertices (words) of B by the elements of NLS as: $\forall c \in V_C, \lambda(c) \in T_C \times I \cup \{*\}$ and to avoid the ambiguity of relation node we only consider such $r \in V_R$, such that $\lambda(r) \in T_R^2 \cup T_R^*$ if and only if $l_i(r, w_i) = 1$.*

$\lambda(r)$ are determined by the various levels of relation arity tree, where r is the relationship between the words in the context of the given string and $\lambda(c)$ are determined by the levels of concept type hierarchy tree, where c is a word of a given string.

Finally we define a CG representing a parse tree of a given string (sentence) as

$$CG = (B, NLS, \lambda)$$

3.1 Concept Type Hierarchy and Relation arity

Conceptual graphs may assert episodic information about particular individuals, or they may express general principles in the semantic network [4]. According to John F. Sowa [4], any representation must satisfy the following constraints:

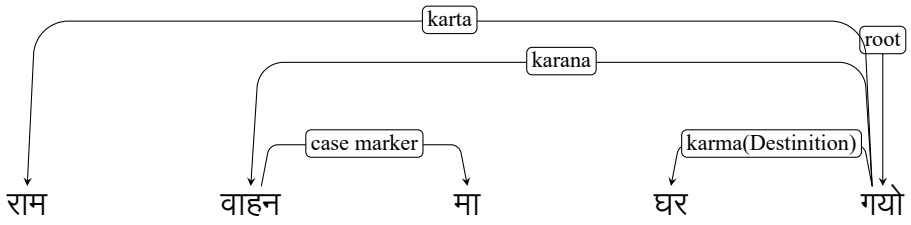

Figure 3: Linear Dependency

- "Connectivity: The algorithms for language parsing, generation, and reasoning depend on the ability to start from any concept and traverse the entire graph. implementation must support some form of forward and backward pointers linking all the nodes".

- "Generality: Although most primitive conceptual relations are dyadic, the formalism allows relations with any number of arcs. Furthermore, any concept may have any number of relations attached to it, and the number may increase as more assertions are made. The implementation must support all these options".

- "No privileged nodes: Any concept in a conceptual graph may be treated as the head. The choice of concept to express as a subject or predicate depends on focus and emphasis, but the representation should not presuppose one choice of root or head (as trees and frames typically do)".

- "Canonical formation rules: The four rules of copy, restrict, join, and simplify are used throughout the system in reasoning and parsing. The implementation must make these operations fast and simple".

Concept type nodes are the central directories for semantic attribute about a concept type. We consider part of speech for each such nodes, as in conceptual dependency and dependency parsing, the verb plays a central role in the structure. Relation type records specify semantic information about a conceptual relation type; i.e., they specify the inter-relationship between two concept type nodes. In our context all the relation types are of arity two as each karaka can only links a demand node to a source node. Concept and Relation Hierarchy are depicted in figure 2 & 3 respectively. T_C consists of classes and sub-classes of part of speech. In case of relation nodes we are considering two types of relationship viz.

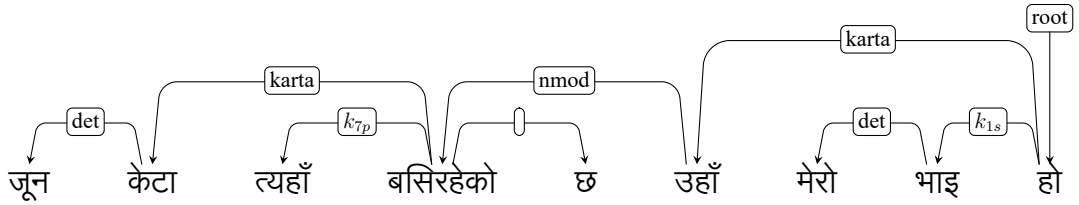

Figure 4: Linear Dependency

1. first we consider karakas as the relation attribute between the concept nodes according to the karaka chart for each demand and source node with respect to the formal grammar rules.

2. secondly we consider all types of generic relations which links two or more non-verb concept nodes.

The first kind relation types are of arity two as each karaka can only links a demand node to a source node.

3.2 Implementation

In this section we describe and implement a conceptual graph as a parse tree for a given string (sentence). Sowa [4] introduced the CG describing a semantic interpreter that started with a parse tree then generating a CG representing the meaning of the sentence. In our study we are describing a CG defined as

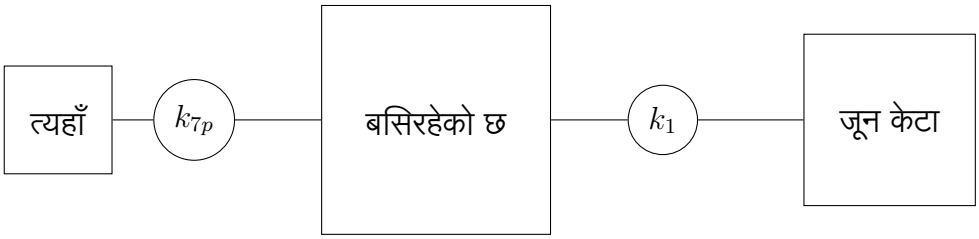

Figure 5: CG_1 for the sentence S_1.

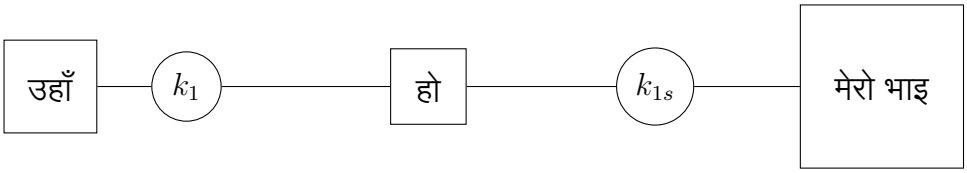

Figure 6: CG_2 for the sentence S_2.

$CG = (B, NLS, \lambda)$ as a parsing of the given sentence, with modified definition of λ (def. 5) to avoid the multiple relation between the same two words. For this purpose we consider a projective sentence "राम= Ram वाहनमा= in vehicle घर= home गयो"= went. We construct a problem bipartite graph as mention in section 2 and then obtain an ordered bipartite graph (definition 3) as depicted in figure 3. Members or elements of the NLS is from the relation arity tree (figure 2) and the Concept Type Hierarchy tree (figure 1). To identify the generic and individual marker we consider the syntax and the semantics of the natural language. Finally applying λ to B we get a CG as a parsing of the given sentence as depicted in figure 7. The conceptual graph in Figure 7 is a sorted version of logic representing a parse. Each of the four concepts have a type label, which refers to: राम, वाहनमा, घर and गयो. One of the concepts is an individual marker. Each of the three conceptual relations has a type label that represents the type of relation: karta (k_1), destination (k_{2p}), or karana (k_3). The CG as a whole demonstrate that the person राम is the agent of some instance of गयो, घर is the destination, and वाहनमा is the instrument.

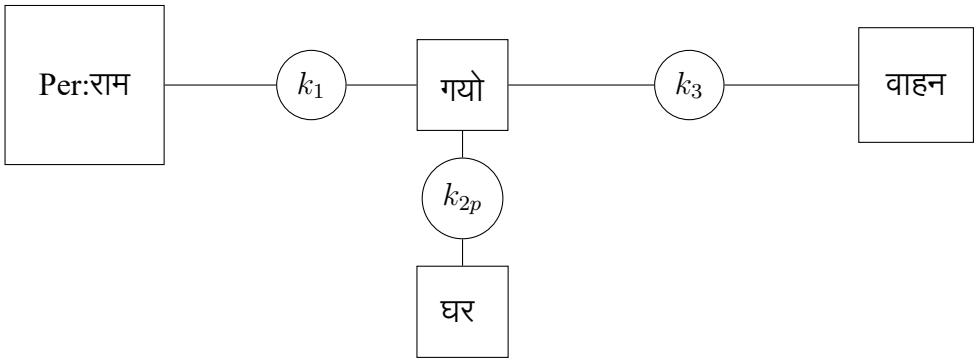

Figure 7: Conceptual Graph as a Parsing of the given sentence.

Figure 7 can be translated to the Sowa's [11] formula as:
$(\exists x)(\exists y)$(गयो(x)∧ Person (राम)∧घर(y)∧k_1(x,राम)∧k_{2p}(x,घर)∧k_3(x,y))

Theorem 3.4. *Let $S_1, S_2 \subseteq X$, such that $X = S_1 \cup S_2$ where X is a non-projective string (sentence) and S_1 and S_2 are projective strings, if $\exists$ conceptual graphs CG_1, CG_2 representing parsing S_1 and S_2 respectively, then there exists a conceptual graph CG^* representing a parsing of X such that $CG^* = CG_1 \cup CG_2 \cup e$, where e is a connector relation node between CG_1 and CG_2.*

Non projectivity structure, in contrast to projective dependency are common in all natural language. Similar to that of Hindi non-projectivity is caused in Nepali, due to various linguistic anomaly in the language, such as relative constructions, paired connectives, complex coordinating structures, interventions in verbal arguments by non-verbal modifiers, shared arguments in non-finite clauses, movement of modifiers, presence of case marker etc. To check the validity of the theorem we consider a non−projective sentence S_1 "जून केटा त्यहाँ बसिरहेको छ, उहाँ मेरो भाइ हो" meaning "the boy who is sitting there he is my brother." To analyze the given sentence we first break it into possible chunks as "जून केटा", "त्यहाँ", "बसिरहेको छ", "उहाँ", "मेरो भाइ" and "हो". The linear dependency and verb frame in the context of the sentence can given as follows:
The sentence S can be decompose into two independent projective sentences as

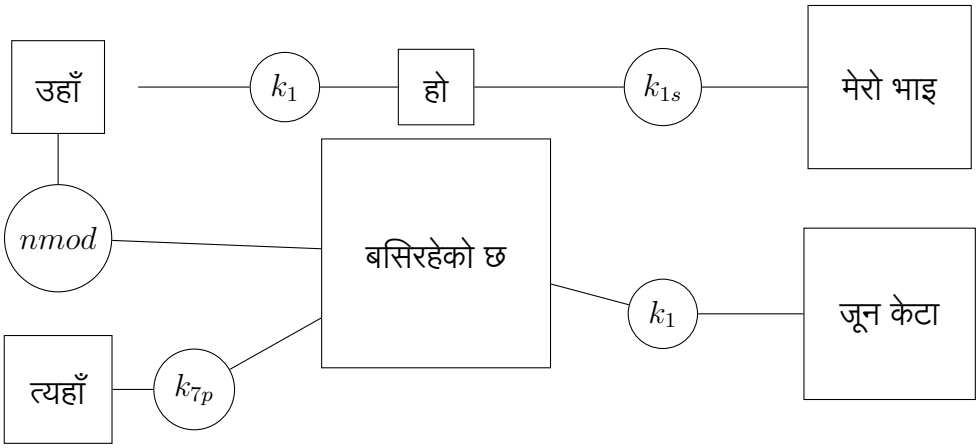

Figure 8: CG^* for the sentence S.

S_1= "जून केटा त्यहाँ बसिरहेको छ" and S_2= "उहाँ मेरो भाइ हो". After constructing the corresponding problem bipartite graphs for both sentences, we apply the definitions $3, 4$, and 5. So we get the Conceptual graphs CG_1 and CG_2 as the parsing of the sentences S_1 and S_2 respectively. The basic requirement to construct a CG^* of a non projective sentence from conceptual graphs of two projective sub-sentences of the given sentence is a modifier connector; i.e., e. For S $e = nmod$, which is a noun modifier. Therefore combining the CG_1 and CG_2 by $e = nmod$ we get the CG^* given by figure 8. Using the same formalism, NL support and λ parsing of the corresponding Gujarati sentences are also obtained.

arc label	**Necessity**	**Vibhakti**	**ConceptType**	**posn**	**reln**
k_1	m	ϕ	Noun	l	c
k_{2p}	m	ϕ	Noun	l	c
k_3	m	मा	Noun	l	c
k_2	O	ϕ	Noun	l	c
k_{7p}	O	ϕ	Noun	l	c

Table 1: Verb Frame in the context of the given sentence.

4 Conclusion

This paper defines a CG model to the parsing of two languages viz. Nepali and Gujarati for the projective sentences and which further can be implemented for many Indian languages with respect to the context of the syntax and semantics of the language. Theorem 1 in section 2 demonstrates that if the Bipartite Graph representing a sentence has a planar embedding then the sentence is projective. Using CG approach for parsing of a natural language has many advantages over other existing methods, the existence of a NL support distinguish and sorts the hierarchy of concepts and relationship between the words. We re-define ordered bipartite graph, NL support in a new context of parsing. The derived mathematical properties could assist future work in research and the development of knowledge representation, in particular, in the area of parsing, which has many applications in Natural Language Processing. We finally propose a theorem to construct a CG as a parse of a non projective sentence by decomposing it into-sub parts (projective).

5 Acknowledgements

The authors acknowledge the support of the Department of Science and Technology, Government of India for funding project entitled "Study and develop a natural language parser for Nepali language" Reference no. SR/CSRI/28/2015(G) under Cognitive Science Research Initiative (CSRI) to carry out this work.

References

[1] John F. Sowa, Conceptual structures: Information processing in mind and machine, 1984.

[2] John F. Sowa, Conceptual Graphs for a Data Base Interface, IBM J. Res. Develop, (1976), $336 - 357$.

[3] Madalina Croitoru, Kees V. Deemter, A Conceptual Graph Approach to the Generation of Referring Expressions, IJCAI, (2007), $2456 - 2461$.

[4] John F. Sowa, Eileen C. Way, Implementing a semantic interpreter using conceptual graphs, IBM J. Res. Develop., **30,** (1986), $57 - 69$.

[5] Rose Dieng, Stefan Hug, Comparison of Personal Ontologies Represented through Conceptual Graphs, ECAI 98, 13th European Conference on Artificial Intelligence, $341 - 345$.

[6] Madalina Croitoru, Bo Hu, Srinandan Dashmapatra, Paul Lewis, David Dupplaw, Liang Xiao, A Conceptual Graph Based Approach to Ontology Similarity Measure, ICCS, (2007), $154 - 164$, .

[7] Bal Krishna Bal, Structure of Nepali Grammar, *Madan Puraskar Pustakalaya*, 2004.

[8] https:/translate.google.co.in

[9] Madalina Croitoru, Ernesto Compatangelo, Conceptual Graph Assemblies, ICCS, 2006.

[10] Akshar Bharti, Veenit Chaitanya, Rajeeb Sangal, Natural Langauage Processing: A Pananain Prospective, Prentice-Hall of India, 1994.

[11] John F. Sowa, Conceptual Graphs, ICCS 99 Proc. 7th International Conference on Conceptual Structures: Standards and Practices, (1999), $1 - 65$.

[12] Madalina Croitoru, Conceptual Graphs at Work: Efficient Reasoning and Applications, Computing Science Dept., University of Aberdeen, 2006.

[13] Douglas B. West, Introduction to Graph theory, Pearson Education, Inc., ISBN-817808-830-4, 2002.