Samāsa-Kartā: An Online Tool for Producing Compound Words using IndoWordNet

Hanumant Redkar
Center for Indian Language Technology,
Indian Institute of Technology Bombay, India
hanumantredkar@gmail.com

Nilesh Joshi
Center for Indian Language Technology,
Indian Institute of Technology Bombay, India
joshinilesh60@gmail.com

Sandhya Singh
Center for Indian Language Technology,
Indian Institute of Technology Bombay, India
sandhya.singh@gmail.com

Irawati Kulkarni
Center for Indian Language Technology,
Indian Institute of Technology Bombay, India
irawatikulkarni@gmail.com

Malhar Kulkarni
Center for Indian Language Technology,
Indian Institute of Technology Bombay, India
malharku@gmail.com

Pushpak Bhattacharyya
Center for Indian Language Technology,
Indian Institute of Technology Bombay, India
pushpakbh@gmail.com

Abstract

Samāsa or compounds are a regular feature of Indian Languages. They are also found in other languages like German, Italian, French, Russian, Spanish, *etc.* Compound word is constructed from two or more words to form a single word. The meaning of this word is derived from each of the individual words of the compound. To develop a system to generate, identify and interpret compounds, is an important task in Natural Language Processing. This paper introduces a web based tool – *Samāsa-Kartā* for producing compound words. Here, the focus is on Sanskrit language due to its richness in usage of compounds; however, this approach can be applied to any Indian language as well as other languages. IndoWordNet is used as a resource for words to be compounded. The motivation behind creating compound words is to create, to improve the vocabulary, to reduce sense ambiguity, *etc.* in order to enrich the WordNet. The *Samāsa-Kartā* can be used for various applications *viz.*, compound categorization, sandhi creation, morphological analysis, paraphrasing, synset creation, *etc.*

1 Introduction

Word compounding is an essential feature of any language. In literature, there are various definitions of the compound word[1]. A compound word is a lexeme that consists of more than one stem. An English compound is a word composed of more than one free morpheme. However, in Sanskrit, a compound, also known as समास (*samāsa*) is defined as पृथगर्थानामेकार्थीभावः समासः (*pṛthagarthā nāmekārthībhāvaḥ samāsaḥ*, placing together two or more words so as to express a composite sense, which is a compound composition)[2]. Example, शिवपत्नी (*śivapatnī*, wife of *śiva* and a benevolent aspect of *devī*) is a *samāsa* or a compound formed from two words शिव (*śiva*, a major divinity in the later Hindu pantheon) and पत्नी (*patnī*, a married woman) which are formed from paraphrase शिवस्य पत्नी (*śivasya patnī*, wife of *śiva* and a benevolent aspect of *devī*). Sanskrit language has high usage of compounds in literature and is rich in producing

[1] http://grammar.ccc.commnet.edu/grammar/compounds.htm
[2] http://lukashevichus.info/knigi/abhyankar_shukla_sans_gram_dic.pdf

compound words. *Pāṇini*, the most referred Sanskrit grammarian, mentioned various types of *samāsa* and compounding system stated in the form of 110 *sutras* (rules) in his grammar book *Aṣṭādhyāyī* (Mishra, 2010).

1.1 Types of *Samāsa* in Sanskrit

In Sanskrit, there are four major types of *Samāsa*:

- **अव्ययीभाव** (***Avyayībhāva***) - In *avyayībhāva samāsa*, first member has primacy[3] (पूर्वपदार्थप्रधान, *pūrva-padārtha-pradhāna*). Here, the first member of this type of nominal compounds is indeclinable, to which another word is added so that the newly formed compound also becomes indeclinable (*i.e.*, अव्यय, *avaya*). Example, यथाशक्ति (*yathāśakti*, in accordance with one's strength).
- **तत्पुरुष** (***Tatpuruṣa***) - In *tatpuruṣa samāsa*, second member has primacy (उत्तरपदार्थप्रधान, *uttara-padārtha-pradhāna*) and the first component is in a case relationship with another. Example, सन्ध्याकालः (*sandhyākālaḥ*, evening time).
- **द्वन्द्व** (***Dvandva***) – In *dvandva samāsa*, both members have primacy (उभयपदार्थप्रधान, *ubhaya-padārtha-pradhāna*). Here, the members are usually noun stems, connected in sense with 'and'. Example, रामलक्ष्मणभरतशत्रुघ्नाः (*rāmalakṣmaṇabharata śatrughnāḥ*, Ram and Laxman and Bharat and Shatrughn).
- **बहुव्रीहि** (***Bahuvrīhi***) - In *bahuvrīhi samāsa*, both members refers to a thing which in itself is not part of the compound (अन्यपदार्थप्रधान, *anya-padārtha-pradhāna*). Example, गजाननः (*gajānanaḥ*, one whose face is that of an elephant).

1.2 IndoWordNet as a Resource

WordNet is a lexical resource composed of synsets and their semantic and lexical relations. Synsets are sets of synonyms or synonymous words (Miller et al., 1990). IndoWordNet[4] is a linked structure of WordNets of major Indian languages from Indo-Aryan, Dravidian and Sino-Tibetan families (Bhattacharyya, 2010).

In this paper, we have taken Sanskrit WordNet[5] as a resource. Sanskrit is an Indo-Aryan language and is one of the ancient languages. It has vast literature and a rich tradition of creating lexica (Kulkarni et al., 2010(a)). The roots of most of the languages in the Indo European family in India can be traced to Sanskrit (Kulkarni et al., 2010(b)). Also, as stated in the article 351 of the constitution of India, the need arises for coining new words when the new object or an action related to it becomes part of the language and gets lexicalized[6]. The grammatical features of Sanskrit are prescribed for use and compounding is an important feature of Sanskrit.

The paper is organized as follows: Section 2 introduces *Samāsa-Kartā* and its components in detail. Section 3 lists the salient features of *Samāsa-Kartā*. Section 4 gives the limitation of *Samāsa-Kartā*. Section 5 describes the related work. Finally, we conclude the paper with the mention of scope and enhancements to this tool and its usefulness in the entire WordNet community.

2 *Samāsa-Kartā*: The Compound Word Producer

2.1 What is *Samāsa-Kartā*?

The *Samāsa-Kartā*[7], also known as Compound Word Producer is an online tool developed to produce compound words. The produced words are formed using rule based system which takes two words from IndoWordNet database (Prabhu et. al, 2012) with the help of IndoWordNet APIs (Prabhugaonkar et. al, 2012). The new word which is produced, is another word, which falls under any of the four types of *samāsas* mentioned above.

There are two types of users for this tool – the lexicographer and the validator. The basic job of lexicographer is to enter words, generate compound words and temporarily add these compound words to the synset in WordNet database. The main task of validator is to validate if the compound words are properly produced and added to the WordNet database.

Samāsa-Kartā basically produces compounds between Noun-Noun (NN-NN), Noun-Adjective

[3] primacy – the fact of being pre-eminent or most important.
[4] http://www.cfilt.iitb.ac.in/indowordnet/
[5] http://www.cfilt.iitb.ac.in/wordnet/webswn/wn.php
[6] http://www.constitution.org/cons/india/p17351.html
[7] http://www.cfilt.iitb.ac.in/wordnet/samaaskarta/

(NN-JJ), Noun-Verb (NN-VM), Adjective-Noun (JJ-NN), Adverb-Noun (RB-NN) pairs. However, it does not deal with the word combinations such as Noun-Adverb (NN-RB), Verb-Verb (VM-VM) and Verb-Noun (VM-NN) as they cannot be compounded.

2.2 Components of *Samāsa-Kartā*

Samāsa-Kartā, the tool, has multiple components which follows the pipeline architecture. Figure 1 shows the block diagram and figure 2 shows the basic interface of the *Samāsa-Kartā*.

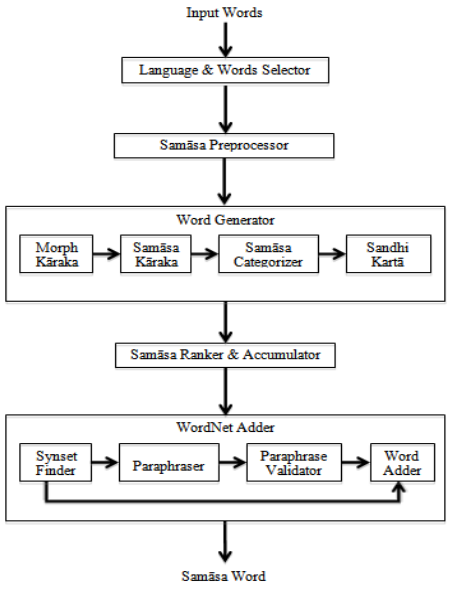

Figure 1. Block diagram of *Samāsa-Kartā*

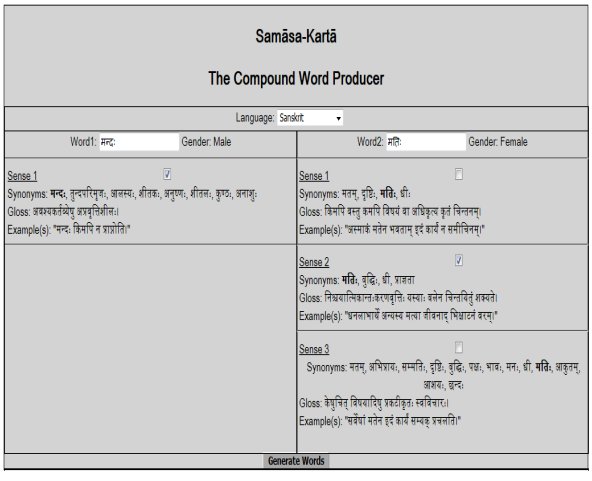

Figure 2. Interface of *Samāsa-Kartā*

The components of *Samāsa-Kartā* are described in detail as follows:

2.2.1 Language and Words Selector

In this module, user selects input language, in our case, Sanskrit and the words are taken from IndoWordNet database to form a compound. Here, the lexicographer types-in any character in the selected input language and all the words in database starting with typed character appear in the drop down list. Once words are selected, their corresponding synset information is displayed accordingly.

For example, if a lexicographer inputs two words, मन्दः (*mandaḥ*, disinclined to work or exertion) as first word and मतिः (*matiḥ*, knowledge and intellectual ability) as second word; we get the following synset information:

For the word मन्दः (*mandaḥ*)

Sense 1

Synonyms: मन्दः (*mandaḥ*), तुन्दपरिमृजः (*tundaparimṛjaḥ*), आलस्यः (*ālasyaḥ*), शीतकः (*śītakaḥ*), अनुष्णः (*anuṣṇaḥ*), शीतलः (*śītalaḥ*), कुण्ठः (*kuṇṭhaḥ*), अनाशुः (*anāśuḥ*)
Gloss: अवश्यकर्तव्येषु अप्रवृत्तिशीलः। (*avaśyakartavyeṣu apravṛttiśīlaḥ*)
Example(s): "मन्दः किमपि न प्राप्नोति।" (*mandaḥ kimapi na prāpnoti*)

For the word मतिः

Sense 1

Synonyms: मतम् (*matam*), दृष्टिः (*dṛṣṭiḥ*), मतिः (*matiḥ*), धीः (*dhīḥ*)
Gloss: किमपि वस्तु कमपि विषयं वा अधिकृत्य कृतं चिन्तनम्। (*kimapi vastu kamapi viṣayaṃ vā adhikṛtya kṛtaṃ cintanam*)
Example(s): "अस्माकं मतेन भवताम् इदं कार्यं न समीचिनम्।" (*asmākaṃ matena bhavatām idaṃ kāryaṃ na samīcinam*)

Sense 2

Synonyms: मतिः (*matiḥ*), बुद्धिः (*buddhiḥ*), धीः (*dhīḥ*), प्राज्ञता (*prājñatā*)
Gloss: निश्चयात्मिकान्तःकरणवृत्तिः यस्याः बलेन चिन्तयितुं शक्यते। (*niścayātmikāntaḥkaraṇa-vṛttiḥ* yasyāḥ balena cintayituṃ śakyate)
Example(s): "धनलाभार्थे अन्यस्य मत्या जीवनाद् भिक्षाटनं वरम्।" (*dhanalābhārthe anyasya matyā jīvanād bhikṣāṭanaṃ varam*)

Sense 3

Synonyms: मतम् (*matam*), अभिप्रायः (*abhiprāyaḥ*), सम्मतिः (*sammatiḥ*), दृष्टिः (*dṛṣṭiḥ*), बुद्धिः (*buddhiḥ*), पक्षः (*pakṣaḥ*), भावः (*bhāvaḥ*), मनः (*manaḥ*), धी (*dhī*), मतिः (*matiḥ*), आकुतम् (*ākutam*), आशयः (*āśayaḥ*), छन्दः (*chandaḥ*)

Gloss: केषुचित् विषयादिषु प्रकटीकृतः स्वविचारः। (*keṣucit viṣayādiṣu prakaṭīkṛtaḥ svavicāraḥ*)

Example(s): "सर्वेषां मतेन इदं कार्यं सम्यक् प्रचलति।" (*sarveṣāṃ matena idaṃ kāryaṃ samyak pracalati*)

Here, the lexicographer chooses sense 1 of the word मन्दः (*mandaḥ*) and sense 2 of the word मतिः (*matiḥ*) to form a compound word मन्दमति (*mandamati*, lacking intelligence). He/she also has freedom to select/deselect the synonymous words of these selected synsets. Also, in this module, the proper care has been taken to avoid words which cannot form *samāsa*. There are some words having specific case endings which cannot be compounded, *e.g.,* a word यथा (*yathā*, in which manner) can be compounded; however its synonyms यत्प्रकारेण (*yatprakāreṇa*), येन_प्रकारेण (*yena_prakāreṇa*) cannot be compounded, as they are specific case ending adverbs.

After selecting the appropriate synset and its synonyms, lexicographer finally proceeds to generate *samāsas* or compound words. The compound words are then processed using the following modules.

2.2.2 *Samāsa* Preprocessor

The *Samāsa* Preprocessor performs a check whether the input words are valid to form a *samāsa* or not. Here, it will check part-of-speech (POS) of each input word and validates if the combinations of POS like NN-NN, NN-JJ, JJ-NN, RB-NN, *etc.* can be formed.

2.2.3 Word Generator

The Word Generator internally processes each input word by using *Morph-Kāraka, Samāsa-Kāraka, Samāsa* Categorizer and *Sandhi-Kartā* to form a compound word. The details of these sub modules are as follows:

Morph Kāraka

Morph-Kāraka or Morph Analyzer is executed once the *Samāsa* Preprocessor provides it the validated input words. In this module, each input word is taken and converted to its root form by applying standard morphological rules. This is required, as in Sanskrit WordNet, all nouns are stored in nominative singular form. In order to make compound of these words, we need to bring these nouns to their root form. Table 1 illustrates some of the words processed through *Morph-Kāraka*.

स्वरान्त-शब्दाः (*svarānta-śabdāḥ*) (vowel-ending words)		**व्यञ्जनान्त-शब्दाः** (*vyañjanānta-śabdāḥ*) (consonant-ending words)	
अकारान्त (*akārānta*)	मन्दः → मन्द (*mandaḥ → manda*)	**चकारान्त** (*cakārānta*)	वाक् → वाच् (*vāk → vāc*)
आकारान्त (*ākārānta*)	विद्या → विद्या (*vidyā → vidyā*)	**जकारान्त** (*jakārānta*)	भिषक् → भिषज् (*bhiṣak → bhiṣaj*)
इकारान्त (*ikārānta*)	मतिः → मति (*matiḥ → mati*)	**तकारान्त** (*takārānta*)	भगवान् → भगवत् (*bhagavān → bhagavat*)
ईकारान्त (*īkārānta*)	नदी → नदी (*nadī → nadī*)	**दकारान्त** (*dakārānta*)	शरद् → शरद् (*śarad → śarad*)
उकारान्त (*ukārānta*)	भानुः → भानु (*bhānuḥ → bhānu*)	**नकारान्त** (*nakārānta*)	आत्मा → आत्मन् (*ātmā → ātman*)
ऋकारान्त (*ṛkārānta*)	माता → मातृ (*mātā → mātṛ*)	**सकारान्त** (*sakārānta*)	तेजः → तेजस् (*tejaḥ → tejas*)

Table 1. Words processed through *Morph-Kāraka*

Once the morphological analysis is done on input words, they are given to *Samāsa-Kāraka* for further processing.

Samāsa-Kāraka

The *Samāsa-Kāraka* takes the processed words from *Morph-Kāraka* and applies standard *samāsa* rules based on grammar. The *Samāsa-Kāraka* works at the semantic as well as syntactic level. At semantic level, meanings of the words are considered from the gloss to form the compounded word. At syntactic level, the inflections are appended/not appended to the morphed words. The processed words along with its *Samāsa* type are passed to the *Samāsa* Categorizer as an input.

For example,

1) आत्मन् + शक्ति (*ātman + śakti*) – Here, *Samāsa-Kāraka* identifies that both the words आत्मन् (*ātman*) and शक्ति (*śakti*) follows 2.2.8 rule षष्ठी (*ṣaṣṭhī*) *of Pāṇinian* grammar. Hence, आत्मन् (*ātman*) is eligible to form *Samāsa* with the

शक्ति (*śakti*). However, the rule number 8.2.7 नलोपः प्रातिपदिकान्तस्य (*nalopaḥ prātipadikān tasya*) *of Pāṇinian* grammar says that the न् (*n*) should be removed from the word आत्मन् (*ātman*). Hence, words आत्म (*ātma*) and शक्ति (*śakti*) is sent to *Samāsa* Categorizer for further processing.

2) देव + ईश (*deva* + *īśa*) – Here, there is no infection, hence these words are directly passed to the *Samāsa* Categorizer.

Samāsa Categorizer

Samāsa Categorizer identifies category of a *samāsa* like *Avyayībhāva*, *Tatpuruṣa*, *Dvandva* & *Bahuvrīhi* as per the *samāsa* rules. Further, it identifies its sub categories. It generates paraphrased information using gloss of input words. This paraphrased information is stored here, which is further used in the WordNet Adder for paraphrasing of compound words.

Sandhi-Kartā

Sandhi-Kartā or Sandhi Joiner helps in joining two words together which are passed through *Samāsa* Categorizer. The words are joined together by following sandhi rules of the language into consideration. The *Sandhi-Kartā* performs on all the combinations of the selected synset words and produces list of joined words. All these joined words are given to the *Samāsa* Ranker & Accumulator module.

Some of the examples of *Sandhi-Kartā* usage for words in Sanskrit are illustrated in table 2.

2.2.4 *Samāsa* Ranker and Accumulator

In this module, all the combinations of words are ranked and accumulated together as per the most frequent usage of words in the original WordNet synsets. Here, *Samāsa* Ranker Algorithm is used to rank the accumulated *samāsas.* Once the ranking and accumulating of words are done, the *samāsas* will be passed through the WordNet Adder module where its validity is checked and added to the WordNet.

2.2.5 WordNet Adder

WordNet Adder is a semi-automatic process where newly formed *samāsas* are passed through sequence of steps before adding to the synset in the WordNet.

स्वरान्त-शब्दाः (*svarānta-śabdāḥ*) (vowel-ending words)	
अकारान्त (*akārānta*) (words ending with a)	देव + ईश → देवेश (*deva + īśa → deveśa*)
आकारान्त (*ākārānta*) (words ending with ā)	विद्या + आलय → विद्यालय (*vidyā + ālaya → vidyālaya*)
इकारान्त (*ikārānta*) (words ending with i)	प्रति + उत्तर → प्रत्युत्तर (*prati + uttara → pratyuttara*)
ईकारान्त (*īkārānta*) (words ending with ī)	नदी + ईश → नदीश (*nadī + īśa → nadīśa*)
उकारान्त (*ukārānta*) (words ending with u)	भानु + उदय → भानूदय (*bhānu + udaya → bhānūdaya*)
ऋकारान्त (*ṛkārānta*) (words ending with ṛ)	मातृ + ऋण → मातॄण (*mātṛ + ṛṇa → mātṝṇa*)
व्यञ्जनान्त-शब्दाः (*vyañjanānta-śabdāḥ*) (consonant-ending words)	
तकारान्त (*takārānta*) (words ending with ta)	भगवत् + गीता → भगवद्गीता (*bhagavat + gītā → bhagavadgītā*)
दकारान्त (*dakārānta*) (words ending with da)	शरद् + हविष् → शरद्धविष् (*śarad + haviṣ → śaraddhaviṣ*)
नकारान्त (*nakārānta*) (words ending with na)	आत्मन् + शक्ति → आत्मशक्ति (*ātman + śakti → ātmaśakti*)
सकारान्त (*sakārānta*) (words ending with sa)	मनस् + रथ → मनोरथ (*manas + ratha → manoratha*)

Table 2. Words processed through *Sandhi-Kartā*

Following are the sub modules of WordNet Adder.

Synset Finder

Here, the lexicographer checks if the intended synset already exists in the WordNet. If it exists then the words are directly appended to the intended synset's vocabulary. If the synset does not exist, then it passes through the ***Paraphraser to create gloss of the compound word*** which will help in creating new synset**.**

Paraphraser

The Paraphraser automatically generates most likely gloss of the intended synset on the basis of input words. This gloss or a concept definition of a

synset is given to Paraphrase Validator for further processing.

Paraphrase Validator

Here, the lexicographer checks if the paraphrased gloss is properly generated. If not, it is created / edited manually by using the three principles of synset creation, *viz.*, principle of minimality, coverage and replaceability (Bhattacharyya, 2010). This is given to the Word Adder module.

Word Adder

The lexicographer finally fills-in other synset information like examples, gender, *etc.* and adds to the WordNet using an online synset creation tool - *Synskarta* (Redkar et al., 2014). The resultant *Samāsas* will either be the member of an existing synset or it can be a new synset altogether.

3 Salient Features of *Samāsa-Kartā*

Some of the salient features of *Samāsa-Kartā* are as follows:

- *Samāsa* or compounds are created on the flow.
- *Samāsa* in WordNet helps in identifying meaning or concept of a compound occurring in the literature.
- *Samāsa-Kartā* helps in enriching the standard of the language and to simplify the case-ending words in language under consideration.
- It assists in developing vocabulary, which in turn, helps in improving the word count in a language.
- It helps in automatic generation of paraphrases.
- It helps in compound type identification.
- The compound words produced can be helpful to understand the multi-words.

4 Limitation of *Samāsa-Kartā*

Some of the limitations of *Samāsa-Kartā* are:

- Used only for words in WordNet.
- Possibility of over generation of compounds.
- In Sanskrit, verbs are in its root form; hence word pairs such as VM-VM and RB-VM are not implemented.
- The word combination NN-RB is not possible as adverbs cannot come as a second word in the compound.

5 Related Work

In past, many researchers have worked on compound words, more particularly for Sanskrit Language. To understand the need of the tool presented here, a study is done on different kinds of tools available for usage. Some tools and work which were reviewed in the domain of compound word and its related fields are presented here.

Kumar et al. (2010) presented a Sanskrit compound processor tool, which automatically segments and identifies the type of a compound using the manually annotated data. To understand the compound; their approach involved segmentation, constituency parsing, compound type identification and paraphrasing. This tool can identify the type of compound and suggest its component's root word.

Jha et.al. (2009) proposed an Inflectional Morphology Analyzer for Sanskrit that identifies and analyzes the inflected noun-forms and verb-forms in any sandhi-free text. The tool checks and labels each word as three basic POS categories - *subanta*, *tiṅanta*, and *avyaya*. It is based on a reverse *pāṇinian* approach to analyze *tiṅanta* verb forms into their verbal base and verbal affixes. The methodology used to create database tables to store various morphological components of Sanskrit verb forms is based on the well defined and structured process of Sanskrit morphology described by *Pāṇini* in his *Aṣṭādhyāyī*. This analyzer also includes the analysis of derived verb roots.

Gupta et al. (2009) proposed a Rule Based Algorithm for *Sandhi-Vichedā* of compound Hindi words where one letter (whether single or conjoined) is broken to form two words. Part of the broken letter remains as the last letter of the first word and later part of the broken word forms the first letter of the next letter. A *Sandhi-Vichedā* module breaks the compound word in a sentence into constituent words, which enables to understand the meaning of the words better. This work aids in learning about the language grammar in an easy way.

Satuluri et al. (2013) studied the generation of Sanskrit compounds and rewrote the grammar as a combination of phrase structure rules and the regular grammar. It listed various semantic features as constraints governing the formation of compounds in Sanskrit. The rules taken from *Pāṇini* for compound formation are classified into two sets – the ones which designate a technical term to the input

string or a part thereof termed as *saṃjñāsūtra,* and the others which transform the input string into another termed as *vidhisūtra.* Also, the various semantic information needed by the compound formation rules is stated through ontological approach.

Sanskrit being a highly inflected language in nature, each of its word is inflected. If the words are not used in the correct case-endings, it may lead to a different meaning altogether, giving different context. To simplify the usage of these case-endings, compound words are used. Also, if these compound words are added to the WordNet, it may help in identifying meaning of a compound occurring in the literature.

Hence, we have developed a web based tool called *Samāsa-Kartā*. The approach used in this tool follows the rule-based system which takes two words from IndoWordNet database as an input and produces a compound word. This resultant compound word or *samāsa* can be included as a synset member along with its paraphrase as a gloss in the WordNet.

Work done by Kumar et al. is about identification of compound word, whereas, our tool deals with creation of new compound words. Jha et al. created morphological analyzer; similarly, we have implemented *Morph Kartā* which is created, specifically for WordNet words. Gupta et al., created Sandhi Splitter, however, we have created Sandhi Joiner, which is also specific for *samāsa* of WordNet words. Hence, *Samāsa-Kartā* can be considered as a complete tool of producing compound words related to words in IndoWordNet database.

6 Conclusion

Samāsa is a significant part of most of the languages which is used to express meaning using less number of words. The tool *Samāsa-Kartā*, discussed in this paper, is an attempt to improve upon the richness and coverage of a language using a semi-automated approach. It takes words from IndoWordNet and creates *Samāsa* or compound word(s). *Samāsa-Kartā* uses rule based system to form the compounds by passing through various rules of grammar at each sub module. This tool is able to create new compound words along with its paraphrase which can be added to the WordNet.

7 Future Scope and Enhancements

In future, the tool can be extended to other Indian languages belonging to Indo-Aryan, Dravidian and Sino-Tibetan families *viz.*, Hindi, Marathi, Gujarati, Bengali, Konkani, Kannada, *etc*. It can also be extended to other non-Indian languages like English, German, Italian, *etc.* This tool can have additional features such as non-WordNet words. This will be useful in the light of development of improving the vocabulary of the language, thus enhancing the richness of the language. Some of the major modules of this tool such as *Morph Kāraka*, S*andhi Kartā*, *Samāsa* Categorizer, Paraphraser, etc. can be made available independently.

Acknowledgments

We graciously thank all the members of CFILT[8] lab, IIT Bombay for providing necessary help and guidance needed for the development of *Samāsa-Kartā.* Further, we sincerely thank the members IndoWordNet and Global WordNet community.

References

Amba Kulkarni, Soma Paul, Malhar Kulkarni, Anil Kumar , Nitesh Surtani : Semantic Processing of Compounds in Indian Languages, Proceedings of COLING 2012, Mumbai, December 2012.

Anil Kumar, Vipul Mittal and Amba Kulkarni: *Sanskrit Compound Processor*, Sanskrit Computational Linguistics - 4th International Symposium, New Delhi, India, 2010.

George Miller, R., Fellbaum, C., Gross, D., Miller, K. J. 1990. *Introduction to wordnet: An on-line lexical database.* International journal of lexicography, OUP. (pp. 3.4: 235-244).

Girish Nath Jha, Muktanand Agrawal, Subash, Sudhir K. Mishra, Diwakar Mani, Diwakar Mishra, Manji Bhadra, Surjit K. Singh, *Inflectional Morphology Analyzer for Sanskrit*, Sanskrit computational linguistics. Springer Berlin Heidelberg, 2009.

Hanumant Redkar, Jai Paranjape, Nilesh Joshi, Irawati Kulkarni, Malhar Kulkarni, and Pushpak Bhattacharyya. 2014. *Introduction to Synskarta: An Online Interface for Synset Creation with Special Reference to Sanskrit.* ICON 2014, Goa, India.

[8] http://www.cfilt.iitb.ac.in/

Malhar Kulkarni, Chaitali Dangarikar, Irawati Kulkarni, Abhishek Nanda and Pushpak Bhattacharyya. 2010(a). *Introducing Sanskrit Wordnet*. In Principles, Construction and Application of Multilingual WordNets, Proceedings of the 5th GWC, edited by Pushpak Bhattacharyya, Christiane Fellbaum and Piek Vossen, Narosa Publishing House, New Delhi, 2010, pp 257 – 294.

Malhar Kulkarni, Irawati Kulkarni, Chaitali Dangarikar and Pushpak Bhattacharyya. 2010(b). *Gloss in Sanskrit Wordnet*. In Proceedings of Sanskrit Computational Linguistics. Jha. G. Berlin: Springer-Verlag / Heidelberg. pp 190-197.

Neha R. Prabhugaonkar, Apurva S. Nagvenkar, and Ramdas N. Karmali. 2012. *IndoWordNet Application Programming Interfaces.* In 24th International Conference on Computational Linguistics (COLING 2012), p. 237.

Pavankumar Satuluri, Amba Kulkarni, *Generation of Sanskrit Compounds*, Proceedings of ICON, 2013.

Priyanka Gupta, Vishal Goyal. 2009. *Implementation of Rule Based Algorithm for Sandhi-Vicheda of Compound Hindi Words*. JCSI International Journal of Computer Science Issues, Vol. 3, 2009.

Pushpak Bhattacharyya. 2010. *IndoWordNet*. In the Proceedings of Lexical Resources Engineering Conference (LREC), Malta.

Ramashankar Mishra. 2010. *अष्टाध्यायीसूत्रपाठः*. Motilal Banarasidas publishers pvt. ltd, New Delhi (ISBN 978-81-208-2748-6).

Venkatesh Prabhu, Shilpa Desai, Hanumant Redkar, Neha Prabhugaonkar, Apurva Nagvenkar, Ramdas, Karmali. 2012. *An Efficient Database Design for IndoWordNet Development Using Hybrid Approach*. COLING 2012, Mumbai, India. p 229.