Lexico-Semantic Relations for Punjabi WordNet

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering
in
Computer Science and Engineering

Submitted By
Ashish Narang
(Roll No. 801032003)

Under the supervision of:
Mr. Parteek Bhatia
Assistant Professor

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004

June 2012

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, "*Lexico-Semantic relation for Punjabi WordNet*", in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Computer Science and Engineering* submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Mr. Parteek Bhatia* and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

(Ashish Narang)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

(Mr. Parteek Bhatia)
Assistant Professor
Computer Science and Department
Thapar University, Patiala

Countersigned by

(Dr. Maninder Singh)
Head
Computer Science and Engineering Department
Thapar University
Patiala

(Dr. S. K. Mohapatra)
Dean (Academic Affairs)
Thapar University
Patiala

Acknowledgement

First of all, I would like to express my gratitude towards Thapar University**,** for providing me a platform to do my thesis work at such an esteemed institute.

I wish to express my deep gratitude to Mr. Parteek Bhatia, Assistant Professor, Computer Science and Engineering Department, Thapar University, Patiala for agreeing to become my mentor and aiding me with my research. The crucial guidance and direction that I received has bestowed upon me the skills needed to successfully complete my thesis.

I would like to thank Dr. Maninder Singh, Head, Computer Science and Engineering Department, Thapar University, Patiala who has been a constant source of inspiration for me throughout this work.

I am also thankful to all the staff members of the Department for their full cooperation and help.

I am Thankful to my family and all my friends for their blessings and moral support. Thanks for boosting me with constant encouragement, support and confidence.

Finally, I would like to thank God for providing me with the strength and ability to complete my work.

Ashish Narang

Abstract

WordNet is a lexical structure composed of synsets. Synsets are linked by different lexical and semantic relations like hypernymy, hyponymy, meronymy, troponymy, holonymy, entailment, antonym and gradation *etc.* India is a multilingual country where people speak many different languages, which results in communication barrier. This acted as a motivation behind the building of IndoWordNet. IndoWordNet is a linked structure of WordNets of major Indian languages from Indo-Aryan, Dravidian and Sino-Tibetan families. Many Indian languages are using the expansion approach to develop their WordNets from the Hindi WordNet. Punjabi is the language of hundreds of millions of people in India, and is the religious language of all Punjabis around the world. Surprisingly, little work has been done in the field of computerized language and lexical resources for this language. It is therefore worthy to build up a Punjabi lexical resource (WordNet) that can discover the richness of Punjabi language.

This thesis comprises of six chapters. Chapter 1 provides the brief overview of WordNet, challenges in natural language processing, role of WordNet in natural language processing, basic WordNet principle and approaches for WordNet creation. Chapter 2 describes the various constituents of WordNet and relations used in development of WordNet. The related work done in this field in the form of English WordNet and Hindi WordNet are also presented in this chapter. In chapter 3 problem statement is specified along with the objectives defined for thesis. A methodology to achieve those objectives is also given in chapter 3. Chapter 4 describes approach towards building a lexical resource in Punjabi language. Punjabi WordNet is being developed from Hindi WordNet using the expansion approach. The process of creation of synsets for Punjabi WordNet from Hindi WordNet, proposed database design for storage of lexical and semantic relations, porting of text files to designed database are also presented in this chapter. Algorithms to create the lexical relations like antonymy, gradation, compounding and conjunction *etc*. for Punjabi WordNet have also been discussed in this chapter. Chapter 5 demonstrates the working of various tools like lexical creation tool and web application for Punjabi WordNet. The conclusion and future scope of work presented in this thesis is described in chapter 6. Different tools for the creation of lexical relations like antonymy, compounding,

conjunction, gradation and web application for Punjabi WordNet has been developed during this thesis work.

Keywords: *WordNet, Semantic relations, Lexical relations, Hypernymy, Hyponymy.*

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

The growing amount of non structured information, stored in natural language, requires the availability of computational systems able to handle this type of information. This has in turn encouraged the development of technologies that would solve this problem and can help in accessing this information more efficiently and quickly. Natural language processing (NLP) provides tools and techniques that can allow the implementation of natural language-based interfaces to computer systems that can enable communication between man and machine in natural languages [6]. These techniques also enable people to organize, extract and use the knowledge contained in these huge collections of natural language electronic data. Examples of Language Technology (LT) applications include Machine Translation (MT), Information Extraction (IE), Information Retrieval (IR), document classification and summarization, speech recognition and synthesis *etc.* Lexical resources have become important basic necessity for NLP attributes. WordNet is one of the lexical resource for NLP.

WordNet is a large lexical database of language. In WordNet, words are grouped together according to their similarity of meanings. WordNet maintains concepts in a language, relations between concepts and their ontological detail. Each concept in a language represents a synset. Synsets are basic building blocks of WordNet. Synset is composed of gloss, example sentences and set of synonym words that are used for the concept. Besides synset data, a WordNet maintains lexical and semantic relations. Lexical relations like antonomy and gradation are between the words in a language whereas semantic relations like hypernymy, hyponymy, meronymy, holonymy, entailment, troponymy and casuation are between concepts in a language. WordNet structure makes it a useful tool for computational linguistics and natural language processing. The major applications of WordNet are text categorization, text summarization, word sense disambiguation, machine translation and crossword puzzle generation *etc.*

The various challenges in the field of natural language processing, role of WordNet in natural language processing, basic principle of WordNet and different approaches for the creation of WordNet are further described in this chapter.

1.1 Natural language processing

Natural language processing is a subfield of artificial intelligence and linguistics. It studies the problems of automated generation and understanding of natural human languages. Natural language generation systems convert information from computer databases into human language, and natural language understanding systems convert samples of human language into more formal representations that are easier for computer programs to manipulate [6]. NLP has significantly overlapped with the field of computational linguistics, and is often considered a sub-field of artificial intelligence.

1.1.1 Challenges in Natural language processing

Natural language processing includes the tasks which deal with human languages. There are various challenges in natural language processing that are to be dealt with. The following are the problems faced while processing human languages.

- **Text categorization**

  Text categorization is the task of assigning a document to one or more classes or categories [11]. For example, text categorization decides whether the document belongs to plants, sports, health and fitness or corporate acquisitions as shown in figure 1.1.

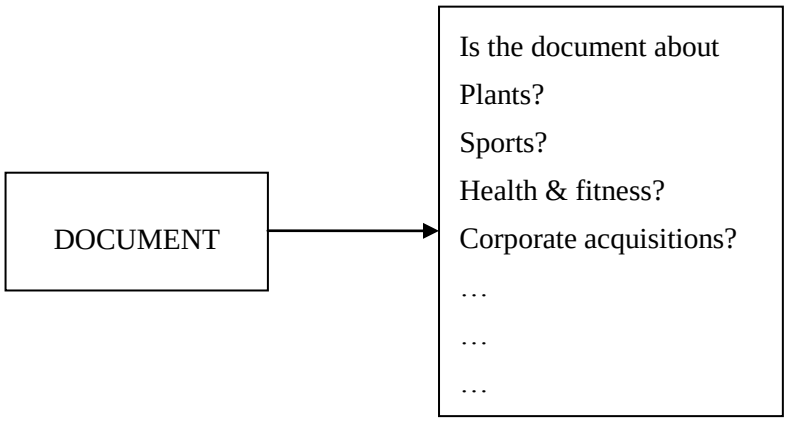

Figure 1.1: Text categorization

- **Sentiment classification**

  A basic task in sentiment analysis is classifying the polarity of a given text document or sentence — whether the expressed opinion in a document or sentence is positive, negative, or neutral as shown in figure 1.2**.**

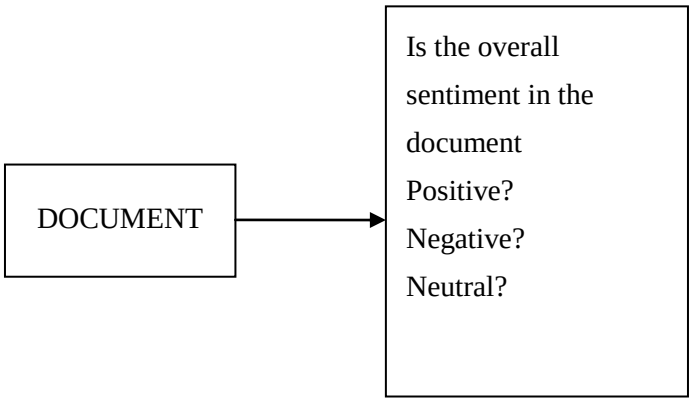

Figure 1.2: Sentiment classification

- **Speech segmentation**

  Speech segmentation is the process of identifying the boundaries between words, syllables, or phonemes in spoken natural languages. In most spoken languages, the sounds which represent successive letters merge into each other, therefore it becomes a difficult task to convert an analog signal to distinct characters. Moreover, in natural speech there are hardly any pauses between successive words, hence if a system has to locate these boundaries then it must be able to consider grammatical and semantic constraints, and also the context [19].

- **Syntactic ambiguity**

  It is a property of sentences which may be reasonably interpreted in more than one way. The grammar for natural languages is ambiguous, *i.e.*, most probably there will be multiple parse trees possible for any given sentence. Thus, semantic and contextual information about a sentence is required in order to choose the most appropriate sense. Specific problem components of syntactic ambiguity include sentence boundary disambiguation. For the structural ambiguity, consider the sentence: "The man saw the girl with a red hat". This sentence is ambiguous as it can be interpreted in two different ways by the machine: The man saw the girl who was wearing a red hat or, the man saw the girl with the help of the red hat. However, the sentence "The man saw the girl with a red hat*"* is not ambiguous for any human reader, one can understand that a hat cannot be used to see, while it is difficult for a computer to interpret the same meaning correctly [20].

1.2 Role of WordNet in NLP

Natural language processing is essential for dealing efficiently with the large quantities of text available online. For example, NLP is used in information retrieval, in text processing and in machine translation. Another essential function is helping the user with query formulation through synonym relationships between words and other hierarchical relationships between concepts [6]. WordNet supports both of these functions. Lexical resources have become fundamental tools within NLP and its related fields. The range of resources available to the researcher is diverse and vast - from simple word lists to complex dictionaries and thesauruses. The resources contain a whole range of different types of explicit linguistic information presented in different formats and at various levels of granularity.

1.2.1 Applications of WordNet in NLP

WordNet is used as a rich source of lexical information in many applications which are discussed below.

- **Word sense disambiguation**

  Word Sense Disambiguation (WSD) is regarded as one of the most interesting and longest-standing problems in Natural Language Processing. It is the process of determining which sense of a word is the intended in a particular context [18]. For example, the word *interest* can mean a charge for borrowing money, or a sense of concern and curiosity. When using language, human rarely stop and consider which sense of a word is intended. For example, consider the sentence "I have an interest in the arts", human reader immediately knows from the surrounding context that interest refers to an appreciation not a charge of borrowing money. But it becomes difficult for software to detect which sense of interest was intended. WSD involves selecting the intended sense of a word for a predefined set of words, with the help of a machine-readable dictionary, such as WordNet. WSD means a task of removing the ambiguity of word in a specific context is important for many NLP applications such as, machine translation and speech processing *etc* [2].

  Uses of Word sense disambiguation

  The word sense disambiguation play vital role for various natural language processing tasks.

- **Machine translation**

  WSD play important role in machine translation. Machine translation is the process of conversion of text from one language called the source language to other language called the target language. WSD is required since the word in source language may have more than one possible translation in the target language.

- **Information Retrieval**

  Information retrieval also benefits from WSD. Ambiguous words in the queries are problematic for information retrieval systems. Hence retrieval engine needs WSD for filtering out the documents with senses irrelevant to the query.

- **Speech Processing**

  Another potential area for WSD is speech processing. In speech synthesis it is important to determine the correct pronunciation of words in order to generate the speech that sounds natural. This process is difficult since there exists some words which are pronounced in more than one way depending upon their meaning. For example, *lead* is used in one way when it is used in the sense "be in front" and another way when it is used in the sense "a type of metal". WSD could help speech synthesis by identifying the correct sense of the word which provides correct pronunciation. The reverse problem may occur in speech recognition for homophones, words that are spelled differently but pronounced in the same way. If different spellings are treated as different senses, WSD can also be helpful in this situation.

- **Text Processing**

  Applications in text processing, grammatical analysis, content and thematic analysis also benefit from WSD techniques.

- **Assessment of semantic similarity**

  It has proved to be essential for a variety of NLP tasks, including syntactic disambiguation, WSD, selection of a suitable translation equivalent, query expansion and document indexing in Information Retrieval (IR). The semantic similarity between two words is calculated on the basis of taxonomical

associations. If we are given two word senses *W1* and *W2*, their similarity is calculated as a function of their belongingness to more general semantic classes [3]. This approach again requires basic lexical and semantic information which is provided by WordNet.

- **To quantify the relatedness of two words**

  Sometimes we wish to measure the relatedness of two word senses, where a word sense is a specific meaning of a particular word. For example, the word *ball* has several senses like, i) it could mean an object used in games ii) a famous dance form iii) a pitch in baseball. A specific method for quantifying how similar two word senses are is known as a measure of semantic relatedness [19].

- **Retrieving words from their meaning**

  Meaning to Word System finds a set of words, closely matching the definition entered by the user. The approach of extracting words from meanings is based on checking the similarity between the user's definition and each entry of the language database without considering any semantics or grammatical information. System makes extensive use of various linguistics resources such as WordNet [9].

- **Audio and video retrieval**

  It is a task of searching the multimedia information from the web relevant to user query. This is a challenge in need of increasingly urgent attention in view of the ongoing development of hypermedia and non-text information [17]. The MultiMediaMiner [28], is a prototype to extract multimedia information and knowledge from the web that uses WordNet to generate conceptual hierarchies for interactive information retrieval.

- **Automatic generation of lexicon**

  A lexicon is the heart of any language processing system. Accurate words with grammatical and semantic attributes are highly desirable for any application like machine translation, information extraction and various forms of tagging or text mining. However, good quality lexicons are difficult to construct requiring enormous amount of time and manpower. Using WordNet, one can automatically generate the dictionary from an input document [25]. The dictionary entries are in the form of Universal Words (UWs) which are

language words (primarily English) concatenated with disambiguation information.

- **Human language technology and artificial intelligence**
  WordNet has become a very useful resource in the human language technology and artificial intelligence. WordNet provides the general lexico-semantic information which helps in open-domain text processing.
- **Trans-lingual applications**
  The development of WordNets in several other languages extends this capability to trans-lingual applications, enabling text mining across languages. For example, in Europe, English WordNet has provided the starting point for the development of a multilingual database for several European languages which is called EuroWordNet project.

1.3 WordNet principle

WordNet has emerged as a crucial resource for NLP tasks. WordNet is an online lexical reference system whose design is inspired by psycholinguistic theories of human lexical memory. WordNet is a large lexical database of language. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms, each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. The resulting network of meaningfully related words and concepts can be navigated with the browser to obtain the information about words.

1.3.1 Lexical matrix

The foundation of WordNet construction is relational semantics. Words and concepts can be looked upon as forming entries in a structure called the Lexical Matrix [16]. Figure 1.3 illustrates lexical matrix. In the lexical matrix rows represent word meanings and columns represent word forms.

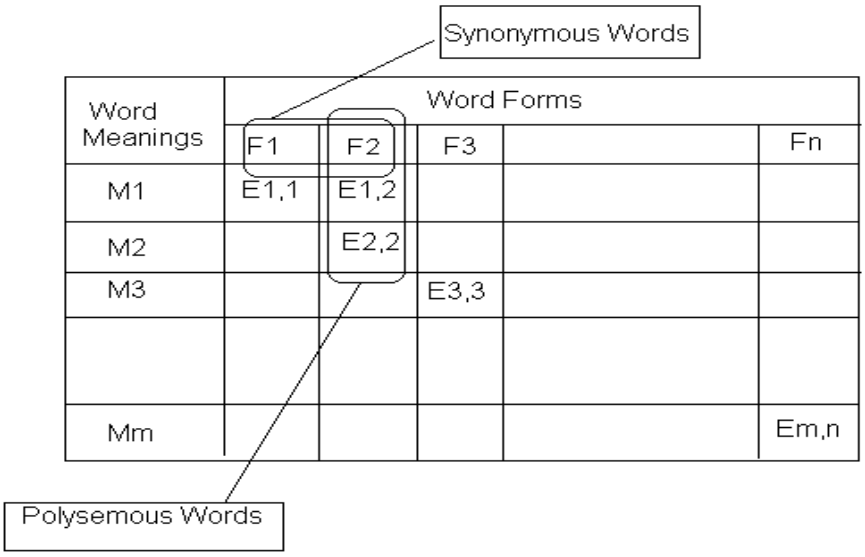

Figure 1.3: Lexical matrix [16]

An entry in a cell of the matrix implies that the form in that column can be used (in an appropriate context) to express the meaning in that row. Thus, entry $E_{1,1}$ implies that word form F_1 can be used to express word meaning *M1*. If there are two entries in the same column, the word form is said to be polysemous. If there are two entries in the same row, the word forms are synonyms (relative to a context) also known as synsets. Mappings between the forms and meanings are many to many: some forms have several different meanings, and some meanings can be expressed by several different forms. For example, for the Punjabi word ਤਰ *tar 'swim, wet, vegetable'* the different meanings and synonyms are shown in figure 1.4.

Polysemous word in column F2

Word Meanings	Word Forms		
	F1	F2	F3
Swim	ਤੈਰਨਾ	ਤਰ	ਤਰਨਾ
Wet		ਤਰ	
Vegetable		ਤਰ	

← Synonyms in the row

Figure 1.4: Lexical matrix for the word ਤਰ

1.4 General methodology for WordNet creation

There are two approaches for the creation of WordNet which are described below.

- **Merge approach**

  In this approach, different senses in which a word can be used are recorded first [4]. Then the lexicographers construct a synset for each of the sense, following the three principles of minimality, coverage and replaceability for synset creation.

- **Expansion approach**

  In the expansion approach, the synsets of the WordNet of a given source language are provided. Each synset is carefully studied for its meaning and accordingly the words of the target language, representing that meaning are collected and put together in a set according to the frequency of their usage.

1.4.1 Comparison of merge and expansion approaches for building WordNet

Both the merge and expansion approaches have their advantages and disadvantages. In the merge approach, there is no distracting influence of another language, which particularly happen when the lexicographer encounters cultural and regional concepts of the source language. The quality of the WordNet will be good using the merge approach, provided the synset maker is well versed with the nuances of the language. But, the disadvantage for this process is that it is very time consuming. In case of expansion approach, the whole WordNet building process becomes well guided in the sense of following the synsets of the source language. Moreover, it has the advantage of being able to borrow the semantic relations of the given WordNet [4]. This can help in saving enormous amount of time. However, the lexicographer can be distracted by synsets which represent cultural and regional concepts. Also there is a problem in finding language specific concepts, *i.e.*, the concepts which are found in particular target language only and are not used in the source language.

There is a predominance of the expansion approach in the WordNet building. This is primarily because many concepts are common across languages. Creating synsets for these universal concepts should be the first step in building a WordNet. If a language has already done this job, it is beneficial to take advantage from this work. Also semantic relations can be borrowed from the source language to be used in target language and it encourages the use of the expansion approach in WordNet building process. If the source and target languages belong to the same language family, the

expansion approach becomes more attractive, since distracting influences of cultural and regional concepts is minimal in this case.

In this chapter, brief overview of WordNet, challenges in NLP, role of WordNet in NLP applications, basic principle of WordNet and different approaches for WordNet creation has been discussed. The principle of creation of synsets, constituents of WordNet, WordNet relations and WordNets available for different languages has been discussed in next chapter.

Chapter 2
Review of Literature

The WordNet project was started in early 1985 by George Miller. In WordNet the words are grouped together according to their similarity of meanings. Two words that can be interchanged in a context are synonymous in that context. For each word there is a synonym set, or synset, in the WordNet, representing one lexical concept. Synsets are the basic building blocks of WordNet. The principle of creation of synsets is discussed in the subsequent section.

2.1 Principle of synset creation

There are three basic principles of minimality, coverage and replaceability which govern the creation of the synsets are described in detail as below.

- **Minimality**: Only the minimal set that can uniquely identify the concept is used to create the synset, *e.g.*, to denote the concept of '*house*' the synset is {ਘਰ *ghar 'room'*, ਮਕਾਨ *makān* '*house'*}. The Punjabi word ਘਰ '*ghar*' is ambiguous and cannot uniquely denote the concept of a '*house*' by itself. For example, it could also mean ਘਰ *ghar* '*house*', ਦੇਸ਼ *'dēsh'* native country or ਪਰਿਵਾਰ '*parivār*' family. The addition of ਮਕਾਨ *makān* '*house'* to the synset brings out this unique sense of house.
- **Coverage**: The synset should contain all the possible words which denote a concept. The words are listed in order of decreasing frequency of their occurrence in the corpus [4]. *e.g.* {ਘਰ *ghar* '*room'*, ਮਕਾਨ *makān* '*house'*}.
- **Replaceability**: The words forming the synset should be mutually replaceable in a specific context. Two synonyms may mutually replace each other in a context *C*, if the substitution of the one for the other in *C* does not alter the meaning of the sentence [4]. For example, {ਸਵਦੇਸ਼ *savdēsh 'motherland'*, ਘਰ *ghar* '*motherland'*}, these two words can be replaced to denote the concept of '*motherland*'.

  ਅਮਰੀਕਾ ਵਿੱਚ ਦੋ ਸਾਲ ਬਤੀਤ ਕਰਨ ਤੋਂ ਬਾਅਦ ਸ਼ਾਮ ਸਵਦੇਸ਼/ਘਰ ਵਾਪਸ ਆ ਗਿਆ *amrīkā vicc dō sāl batīt karan tōṃ bāad shām savdēsh/ghar vāpas ā giā* '*Shyam returned to his motherland after spending two years in America'.*

2.2 Constituent elements of WordNet

WordNet is like a dictionary in that it stores words and meanings. However, it differs from traditional dictionaries in many ways. For example, WordNet attempts to organize information according to the meanings of the words instead of the forms of the words. WordNet divides words into the categories of nouns, verbs, adjectives, and adverbs. Also, WordNet provides position of a word in ontology as an additional feature. The basic constituent elements of the WordNet are described below.

2.2.1 Synset

It is a set of synonymous words. Each synset represents a unique different sense or concept. For example, {ਸਕੂਲ *sakūl* '*school'*, ਪਾਠਸ਼ਾਲਾ *pāṭhshālā* '*school'* } represents the concept of school as *an educational institution*. The words in the synset are arranged according to the frequency of usage.

Different kinds of words found in WordNet are:

- **Polysemous:** Words which have multiple senses are known as polysemous. In WordNet, each word occurs in as many synsets as it has senses. For example, the word ਜੱਗ *jagg 'container, world, feast'* has three senses.
    1. ਸਿਆਣਿਆਂ ਨੇ ਕਿਹਾ ਹੈ ਕੇ ਇਹ ਜੱਗ ਮਿਠਾ ਅੱਗਲਾ ਕਿੰਨ ਡਿੱਠਾ *siāṇiāṃ nē kihā hai kē ih jagg miṭhā agglā kinn ḍiṭṭhā* refers the meaning of ਜੱਗ *jagg* as *world*.
    2. ਪਿੰਡਾਂ ਵਿੱਚ ਕਈ ਵਾਰ ਜੱਗ ਕੀਤਾ ਜਾਂਦਾ ਹੈ ਜਿੱਥੇ ਸਾਰੇ ਰੱਜ ਕੇ ਖਾਂਦੇ ਹਨ *piṇḍāṃ vicc kaī vār jagg kītā jāndā hai jitthē sārē rajj kē khāndē han* explains the concept of word ਜੱਗ *jagg* as feast.
    3. ਪਾਣੀ ਦਾ ਪੂਰਾ ਜੱਗ ਪੀ ਲਓ *pāṇī dā pūrā jagg pī la*. Here, the word ਜੱਗ *jagg* refers to *container*.
- **Monosemous:** Words which can have only one sense are known as monosemous. For example, the word ਰਬ *rab 'God'* has only one sense and hence it will appear in only one synset.
- **Compound Words:** Besides single words, WordNet synsets also sometimes contain compound words which are made up of two or more words but are treated like single words in all respects. For example, WordNet contains the word like ਧਾਰਮਿਕ ਸਥਾਨ *dhārmik sathān* '*religious place*' is made of two words

but treated like a single word. Similarly, the word ਰੀੜ ਦੀ ਹੱਡੀ *rīṛ dī hḍḍī*' *'backbone' etc.* is composed of three words but actually denotes a single word.

2.2.2 Gloss

It describes the concept. It consists of two parts namely text definition and example sentence.

- **Text definition:** It explains the concept denoted by the synset [13]. For example, ਉਹ ਜਗ੍ਹਾ ਜਿੱਥੇ ਵਿਦਿਆ ਦਿੱਤੀ ਜਾਂਦੀ ਹੋਵੇ *uh jaghā jitthē vidiā dittī jāndī hōvē* explains the concept of school as *an educational institution.*
- **Example sentence:** It gives the usage of the words in the sentence [13]. Generally, the words in a synset are replaceable in the sentence. For example, ਇਹ ਸਕੂਲ ਵਿੱਚ ਪਹਿਲੀ ਤੋਂ ਪੰਜਵੀਂ ਤੱਕ ਦੀ ਸਿੱਖਿਆ ਦਿੱਤੀ ਜਾਂਦੀ ਹੈ *ih sakūl vicc pahilī tōṃ pañjvīṃ takk dī sikkhiā dittī jāndī hai* gives the usage for the words in the synset representing school as *an educational institution.*

2.2.3 Position in Ontology

An ontology is a hierarchical organization of concepts, more specifically, a categorization of entities and actions. For each syntactic category namely noun, verb, adjective and adverb, there exists a separate ontological hierarchy [13]. A synset may have multiple parents. The ontology for the synset representing the concept school is shown in figure 2.1.

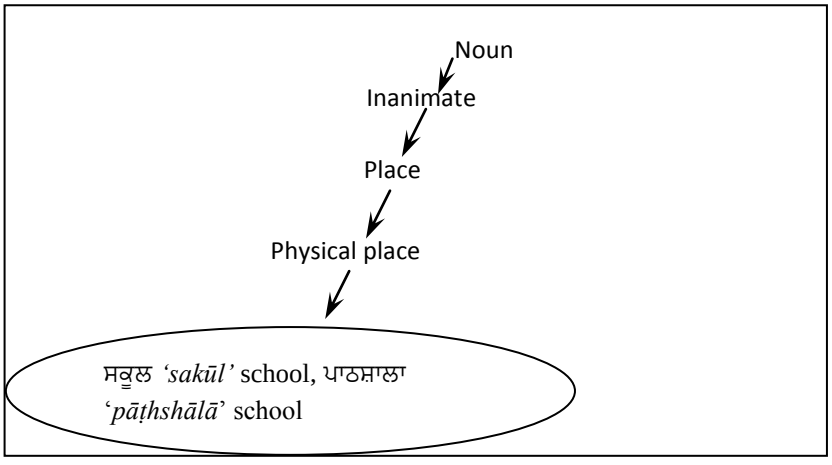

Figure 2.1: Ontology hierarchy [13]

2.3 WordNet relations

WordNet contains the standard information found in dictionaries and thesauri. An additional feature of WordNet is its information about the relationships between words and the synsets. The words and the synsets in the WordNet are linked through two types of relations, *i.e.*, lexical and semantic relations. Lexical relation exists between the word forms while semantic relation exists between the synsets. The different lexical and semantic relations are shown in the figure 2.2.

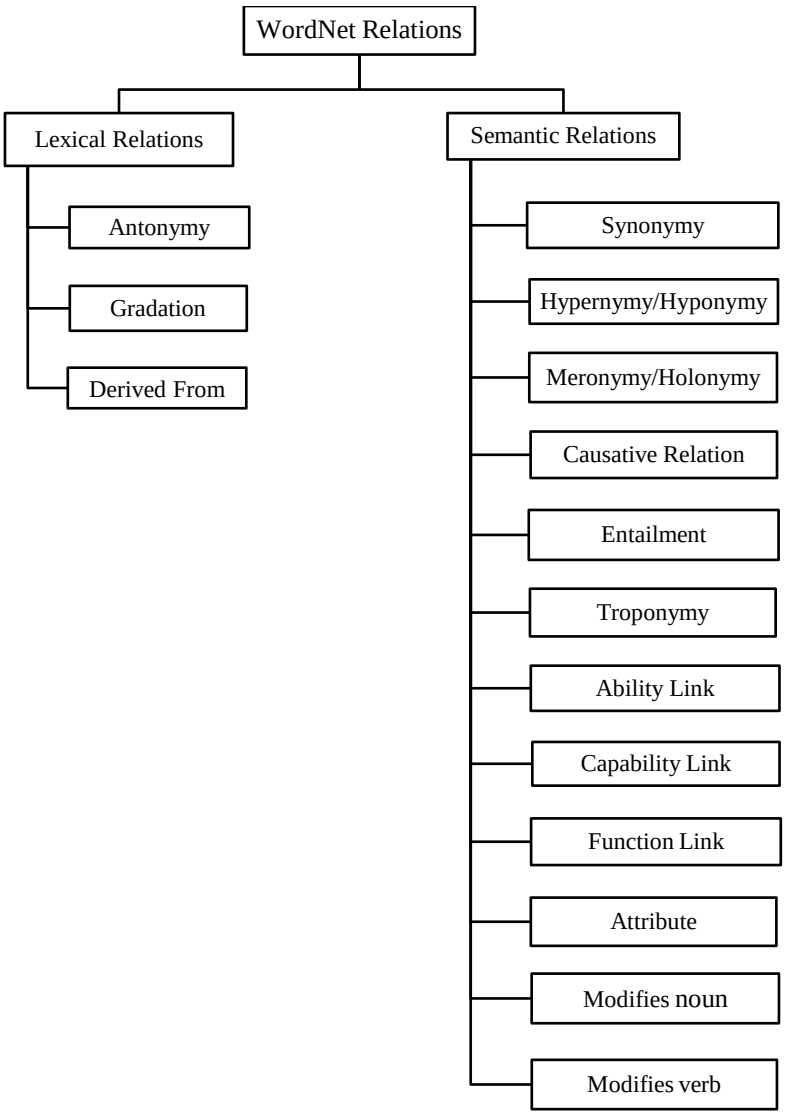

Figure 2.2: WordNet relations

2.3.1 Lexical relations

Lexical relations are the relations between members of two different synsets. For example, consider two synsets given in (1) and (2).

{ਮੋਟਾ *mōṭā 'fat'*, ਭਾਰੀ *bhārī 'fat'*, ਸਥੂਲ *sathūl 'fat'*, ਥੂਲ *thūl 'fat'*, ਨਿੱਗਰ *niggar 'fat'*, ਵਜਨੀ *vajnī 'fat'*}……...……...……(1)

{ਪਤਲਾ *patlā 'thin'*, ਦੁਬਲਾ *dublā thin*,ਕਮਜ਼ੋਰ *kamzōr 'thin'*, ਮਾੜਾ *māṛā 'thin'*}…...…….……..(2)

Here, synsets (1) and (2) are opposites but they does not share antonym relation. Antonym relation exists between two words not between two synsets. Here, the words ਮੋਟਾ *mōṭā 'fat'* and ਪਤਲਾ *patlā 'thin'* are in antonym relation. But, the words ਮੋਟਾ *mōṭā 'fat'* and ਮਾੜਾ *māṛā 'thin'* does not share antonym relation. There are three lexical relations namely antonymy, gradation and derived from.

- **Antonymy**

  Antonymy is a relation that holds between two words that (in a given context) express opposite meanings. It is a lexical relation as it holds between two words and not the entire synset [14]. For example, the word ਨੇੜੇ *nēṛē 'near'* has the antonym as ਦੂਰ *dūr 'far'*.

- **Gradation**

  Gradation is a lexical relation. It represents the intermediate concept between two opposite concepts [14]. For example, the words ਸਵੇਰ *savēr 'morning'*, ਦੁਪਿਹਰ *dupihar 'noon'* and ਸ਼ਾਮ *shām 'evening'* show the concept of gradation. The words ਸਵੇਰ *savēr 'morning'* and ਸ਼ਾਮ *shām 'evening'* are opposites of each other and ਦੁਪਿਹਰ *dupihar 'noon'* is the intermediate concept between these antonyms as illustrated in figure 2.3.

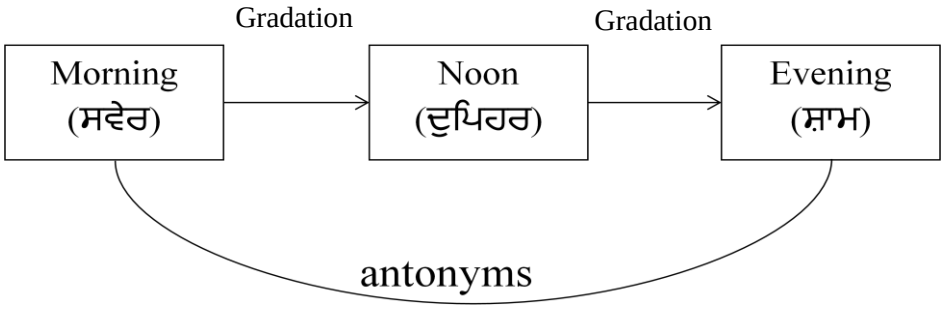

Figure 2.3: Gradation relation

- **Derived From**

  It is a lexical relation that specifies the root form from which a particular word is derived. This relation can go from noun to adjective or vice versa, noun to verb and adjective to verb and aims to handle derivational morphology [13]. For example, ਕ੍ਰਮਵਾਰ *kramvār* '*step by step'* and ਸਿਲਸਿਲੇਵਾਰ *silsilēvār* '*step by step'* are derived from the words ਕ੍ਰਮ *kram 'series'* and ਸਿਲਸਿਲਾ *silsilā* '*series'* respectively.

2.3.2 Semantic relations

Semantic relations are the relations between two whole synsets. For example, hypernym/hyponym is a semantic relation. Consider two synsets given in (3) and (4).

{ਪੌਦਾ *paudā 'plant'*, ਬੂਟਾ *būṭā 'plant'*}……...…… (3)

{ਚਾਹ *cāh 'tea'*}……..…….. (4)

Here {ਪੌਦਾ *paudā 'plant'*, ਬੂਟਾ *būṭā 'plant'*} is hypernym of {ਚਾਹ *cāh 'tea'*} and {ਚਾਹ *cāh 'tea'*} is hyponym of {ਪੌਦਾ *paudā 'plant'*, ਬੂਟਾ *būṭā 'plant'*}. There are total thirteen semantic relations namely hypernymy, hyponymy, meronymy, holonymy, entailment, causation, troponymy, ability link, capability link, functional link, attributes, modifies noun and modifies verb. All the semantic relations are explained with reference to Hindi WordNet document [13].

- **Synonymy**

  Synonymy means similarity of meaning. This relation is used to represent the words that have similar meanings. The relation is symmetric: if '*x'* is similar to '*y'*, then '*y'* is equally similar to '*x'* [14]. For example, the word ਘਰ *ghar 'house'* has synset ਘਰ *ghar 'house'*, ਮਕਾਨ *makān* '*house'* and ਇਮਾਰਤ *imārat 'house'*.

- **Hypernymy/Hyponymy**

  Hypernymy is a semantic relation between two synsets to capture super-set hood. Similarly, hyponymy is a semantic relation between two synsets to capture sub-set hood. The hyponymy relation is transitive and asymmetrical. Hypernymy is the reverse of hyponymy [14]. For example, ਕਬੂਤਰ *kabūtar 'pigeon'* inherits the features from superset ਪੰਛੀ *pañchī 'bird'*. Here ਕਬੂਤਰ

kabūtar 'pigeon' is hyponym of ਪੰਛੀ *pañchī 'bird'* and ਪੰਛੀ *pañchī 'bird'* is hypernymy of ਕਬੂਤਰ *kabūtar 'pigeon'* as shown in figure 2.4

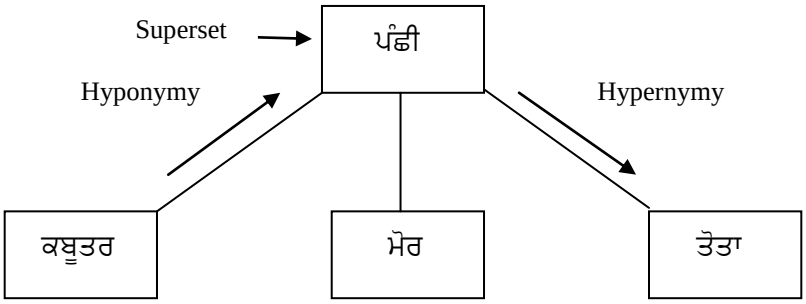

Figure 2.4: Hypernymy/Hyponymy relation

- **Meronymy/Holonymy**

  It is a semantic relation between two synsets. If the concepts '*x'* and '*y'* are related in such a manner that '*x'* is one of the constituent of '*y'*, then '*x'* is the meronym of '*y'* and '*y'* is the holonym of '*x'*. The meronymy relation is transitive and asymmetrical. Holonymy is the reverse of meronymy [14]. For example, ਸਿਰ '*sir' head'* is a part of ਸ਼ਰੀਰ *sharīr 'body'*. ਸਿਰ *sir 'head'* is meronymy ਸ਼ਰੀਰ *sharīr 'body'* and ਸ਼ਰੀਰ *sharīr* '*body'* is holonymy of ਸਿਰ *sir 'head'* as shown in figure 2.5.

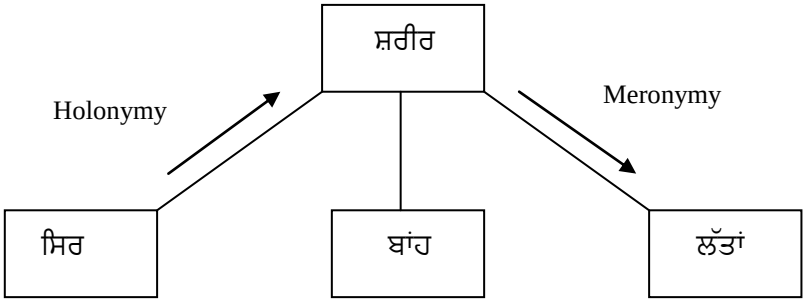

Figure 2.5: Meronymy/Holonymy relation

- **Causative relation**

  In Punjabi, there is a convention of forming causation by making morphological change in the base verb. The causative relation links the causative verbs and the base verbs and show interdependency between them For example, ਖਵਾਉਣਾ *khavāuṇā* '*to make someone to eat*' is a causative verb of ਖਾਣਾ *khāṇā 'eat'*.

- **Entailment**

  It is a semantic relationship between two verbs. A verb '*x'* entails a verb '*y'*, if the meaning of '*y'* follows logically and is strictly included in the meaning '*x'*. The relation of entailment is unilateral, *i.e.*, it is one way relation. For example, ਘਰਾੜੇ ਮਾਰਨਾ *gharāṛē mārnā 'snoring'* entails ਸੋਨਾ *sōnā 'sleeping'* but ਸੋਨਾ *sōnā 'sleeping'* does not entails ਘਰਾੜੇ ਮਾਰਨਾ *gharāṛē mārnā 'snoring'*.

- **Troponymy**

  It is a semantic relation between two verbs when one is specific manner elaboration of another. It shows manner of an action, *i.e.*, '*x'* is a troponym of '*y'* if *to' x'* is *to 'y'* in some manner. For example, ਦਹਾੜਨਾ *dahāṛnā 'to roar'* is troponym of ਬੋਲਨਾ *bōlnā 'to speak'*.

- **Ability link**

  The ability link specifies the inherited features of a nominal concept. For example, {ਤੈਰਨਾ *tairnā 'swim'*} is ability of {ਮਛਲੀ *machlī 'fish'*, ਮੱਛੀ *macchī 'fish'*}.

- **Capability link**

  The capability link specifies the acquired features of a nominal concept. For example, ਤੈਰਨਾ *tairnā 'swim'* is capability of {ਆਦਮੀ *ādmī 'person'*, ਬੰਦਾ *bandā 'person'*}.

- **Function link**

  The relation function link specifies the function of a nominal concept. For example, {ਪੜ੍ਹਾਉਣਾ *paṛhāuṇā 'teach'*,ਗਿਆਨ ਦੇਣਾ *giān dēṇā* '*teach'*} is a function of {ਅਧਿਆਪਕ *adhiāpak 'teacher'*, ਗੁਰੂ *gurū 'teacher'* ,ਮਾਸਟਰ *māsṭar 'teacher'*}.

- **Attribute**

  It shows linkage between noun and an adjective. For example, ਖੰਬਦਾਰ *khambhdār 'having wings'* is an attribute of {ਪੰਛੀ *pañchī 'bird'*, ਪਰਿੰਦਾ '*parindā' 'bird'*}.

- **Modifies noun**

  Certain adjectives can only modify certain nouns. Such adjectives and nouns are linked in the Punjabi WordNet by the relation modifies noun. For example,

{ਸੁਪੁੱਤਰ *suputtar 'son with good character'*, ਚੰਗਾ ਪੁੱਤਰ *caṅgā puttar 'son with good character'*} can be used with {ਆਦਮੀ *ādmī 'person'*, ਬੰਦਾ *bandā 'person'*}.

- **Modifies verb**

  Certain adverbs can only go with certain verbs. Modifies verb is a relation to show connection between such words. For example, the words ਕਦੇ *kadē 'sometimes'* and ਕਿਸੀ ਸਮੇਂ *kisī samēṃ 'sometimes'* can be used with ਕੰਮ ਕਰਨਾ *kamm karnā 'to work'*.

2.4 Related work

WordNet has been developed for different languages starting with English WordNet. Later on, the design of English WordNet was extended to several other WordNet projects. The work that has been done in this field is described further in this section with a brief description of English WordNet, EuroWordNet, Hindi WordNet and IndoWordNet.

2.4.1 English WordNet

English WordNet was created in 1985 and is being maintained by Princeton University under the supervision of psychology professor George A. Miller [16]. The success of English WordNet has inspired several projects that aim at constructing the WordNet for other languages or to develop multilingual WordNets. English WordNet is available online for use. Figure 2.6 depicts the interface of English WordNet showing different senses for the word *mango* as obtained from English WordNet.

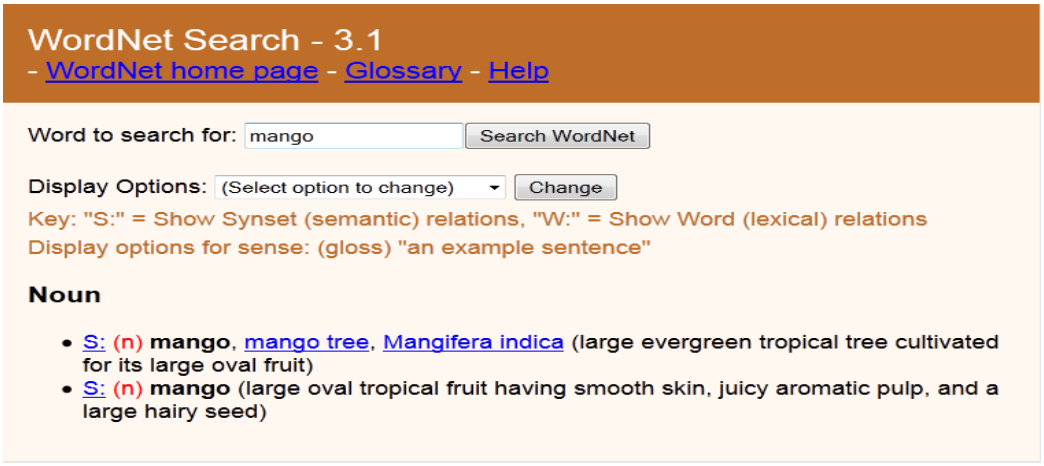

Figure 2.6: Various senses of the word *mango* [8]

The different hypernymy sequences of the word *mango* are shown in figure 2.7. There can be more than one hypernymy sequences for each sense of the word.

Figure 2.7: Hypernymy relations of the word *mango* [8]

2.4.2 EuroWordNet and related projects

EuroWordNet is a system of semantic networks for European languages. Each language develops its own WordNet but they are interconnected with *interlingual links* stored in the *Interlingual Index* (ILI). The original EuroWordNet project dealt with Dutch, Italian, Spanish, German, French, Czech, and Estonian. These WordNets are now frozen, but WordNets for other languages have been developed to varying degrees [26].

BalkaNet WordNet project in 2004 has developed WordNets for Bulgarian, Greek, Romanian, Serbian and Turkish languages [24].

2.4.3 Hindi WordNet

Hindi WordNet is the first WordNet created in India. The design of Hindi WordNet is inspired by famous English WordNet. Hindi WordNet is created and being maintained by Indian Institute of Technology, Bombay under the supervision of Prof. Pushpak Bhattacharya. Hindi WordNet is available online for use. Figure 2.8 shows the web

interface for Hindi WordNet showing different senses of the word आम *aam 'mango'* along with their concepts, example content and category.

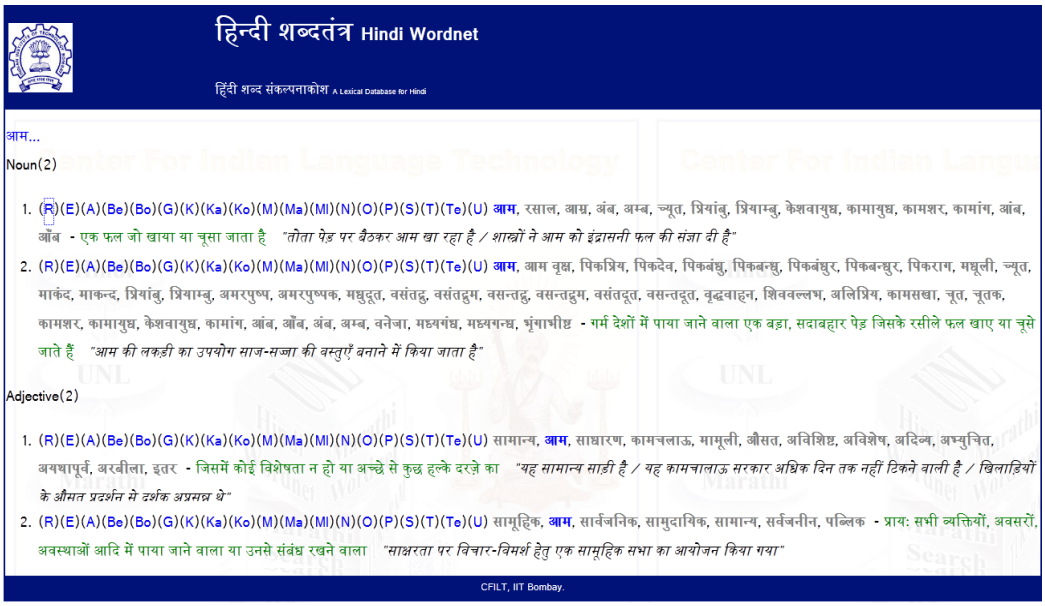

Figure 2.8: Various senses for the word आम [12]

Relations for each word can be obtained by clicking on the (R) link before each sense of the word. Figure 2.9 shows the various hypernymy sequences obtained for the word आम *aam 'mango'*.

Figure 2.9: Hypernymy sequences for the word आम [12]

Later on, Hindi WordNet was extended to Marathi language and various other official languages of India which give rise to a birth called IndoWordNet project.

2.4.4 IndoWordNet

India is a multilingual country where machine translation and cross lingual search are highly relevant problems. These problems require large resources- like WordNets and lexicons- of high quality and coverage. IndoWordNet is a linked structure of WordNets of major Indian languages from Indo-Aryan, Dravidian and Sino-Tibetan families as illustrated in figure 2.10. These WordNets have been created by following the expansion approach from Hindi WordNet which was made available free for research in 2006 [4].

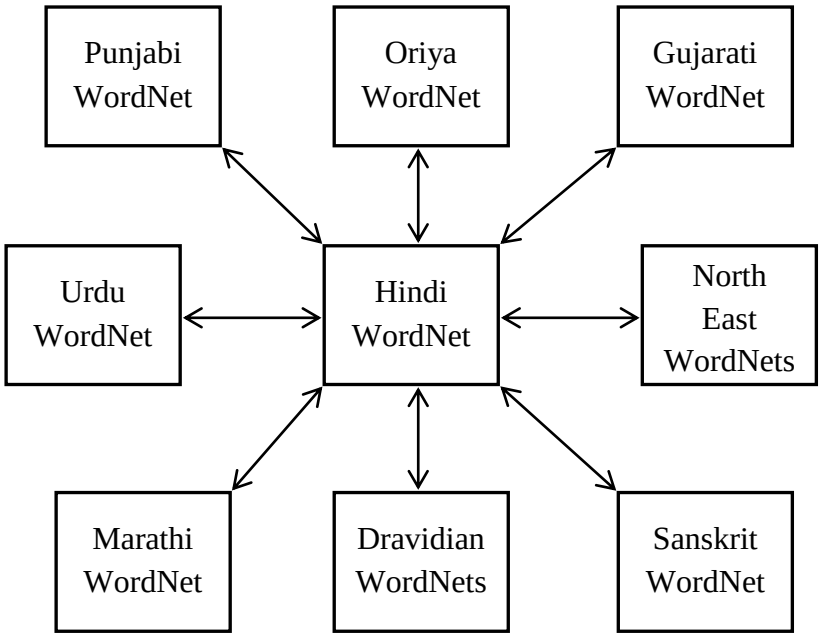

Figure 2.10: Linked IndoWordNet Structure [4]

As a part of IndoWordNet project, Punjabi WordNet is being developed at Thapar University, patiala and Punjabi University, Patiala. The current status of the WordNets being developed for different Indian languages is presented further in this section.

- **Sanskrit WordNet**

  Sanskrit is historically an Indo-Aryan language and one of the 22 official languages of India. It's Motivating to develop Sanskrit WordNet, because for all languages in the Indo European family in India, the roots can be traced to Sanskrit [15]. Sanskrit WordNet is being developed at Indian Insititute of Technology, Bombay. There are 8711 noun synsets, 1454 adjective synsets, 36

adverb synsets and 173 verb synsets entries that have been done for Sanskrit WordNet.

- **Konkani WordNet**

  Konkani language is one of the twenty two languages included in the eighth schedule of the Constitution of India. It is also the official language of the State of Goa. Konakni WordNet is being developed by Goa University, Goa. WordNet for Konkani language has been created using the expansion approach with Hindi as the source language and Konkani as the target language [27]. There are 21567 noun synsets, 5475 adjective synsets, 462 adverb synsets and 2917 verb synsets that have been incorporated in the Konkani WordNet.

- **Gujarati WordNet**

  As a part of IndoWordNet project, WordNet for Gujarati language is being developed from Hindi WordNet using expansion approach [5]. Gujarati, a native language of Indian state of Gujarat, is a member of Indo-Aryan family of languages and it is one of the 22 official languages of India. Gujarati WordNet is being developed at Dharamsinh Desai University, Nadiad. There are 16695 noun synsets, 5790 adjective synsets, 443 adverb synsets and 2801 verb synsets that have been created in the Gujarati WordNet .

- **Bengali WordNet**

  Bengali is one of the major members of the Indo-Aryan family of languages. It is the national language of the country of Bangladesh and also the state language of the province of West Bengal in India [10]. The WordNet for Bengali language is being developed at Indian Statistical Institute, Kolkata, IIT Kharagpur and Jadavpur University, Kolkata. There are 17540 noun synsets, 7245 adjective synsets 586 adverb synsets and 3520 verb synsets that have been created for Bengali WordNet.

- **Bodo WordNet**

  Bodo is a Tibeto-Burman language. This language got Indian govt. recognition as scheduled language from 2003. Bodo WordNet is being developed as expansion approach from Hindi-English WordNet [23] by Guahati University, Assam. There are 3209 adjective synsets, 4921 noun synsets, 1777 verb synsets and 206 adjective synsets that have been created for Bodo WordNet.

- **Urdu WordNet**

  The urdu WordNet is being developed at University or Hyderabad and International Institute of Information Technology, Allahabad [1]. Urdu WordNet is developed from Hindi WordNet using transliterators. There are 209 adverb synsets, 13200 noun synsets 1798 adverb synsets and 3605 adjective synsets that have been created for urdu WordNet.

- **Assamese WordNet**

  Assamese is the easternmost Indo-Aryan language. It is the main spoken language of the state of Assam. This language is recognized as regional language in the eighth schedule of Indian constitution. This language is also used as interstate communication language in many north eastern states specially Arunachal Pradesh and Nagaland [22]. Assamese WordNet is being developed at Guahati University, Assam. There are 2962 adjective synsets, 4390 noun synsets 3530, 174 adverb synsets and 1672 verb synsets that have been created for Assamese WordNet.

- **Punjabi WordNet**

  Punjabi is a member of the Indo-Aryan family spoken by about 91 million people. It is the state language of the province of Punjab in Western India. Punjabi WordNet is being developed by Thapar University, Patiala and Punjabi University, Patiala. There are approximately 8054 noun synsets, 1605 verb synsets, 3568 adjective synsets and 209 adverb synsets have been created for Punjabi WordNet.

2.5 Existing Punjabi WordNet

At present, Punjabi WordNet can be used to find different senses for the word along with their concept, example and category. Moreover, for the input Punjabi word the corresponding synset information can be displayed in Hindi language. Similarly, for the input Hindi word information can be displayed in Punjabi [21].

Figure 2.11 depicts the interface for Punjabi WordNet. Punjabi keypad is used to enter the word which is to be searched.

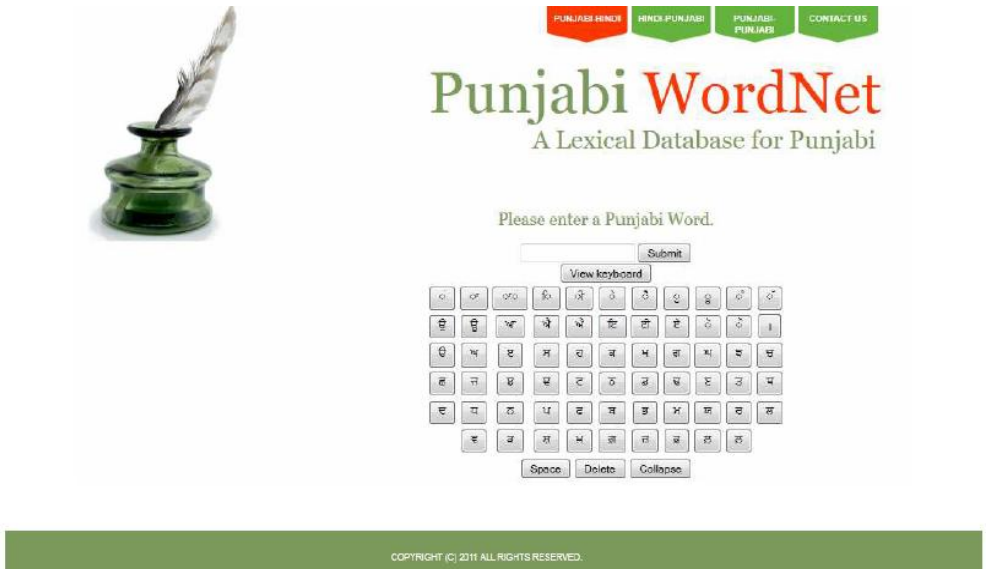

Figure 2.11: Interface for Punjabi WordNet [20]

For example, the user enters a word ਪ੍ਰਤੀਕੂਲਤਾ *pratīkūltā 'incompatible'* from the keypad provided. The system finds the different senses for the word ਪ੍ਰਤੀਕੂਲਤਾ *pratīkūltā 'incompatible'*. Figure 2.12 shows first sense of the word ਪ੍ਰਤੀਕੂਲਤਾ *pratīkūltā 'incompatible'*.

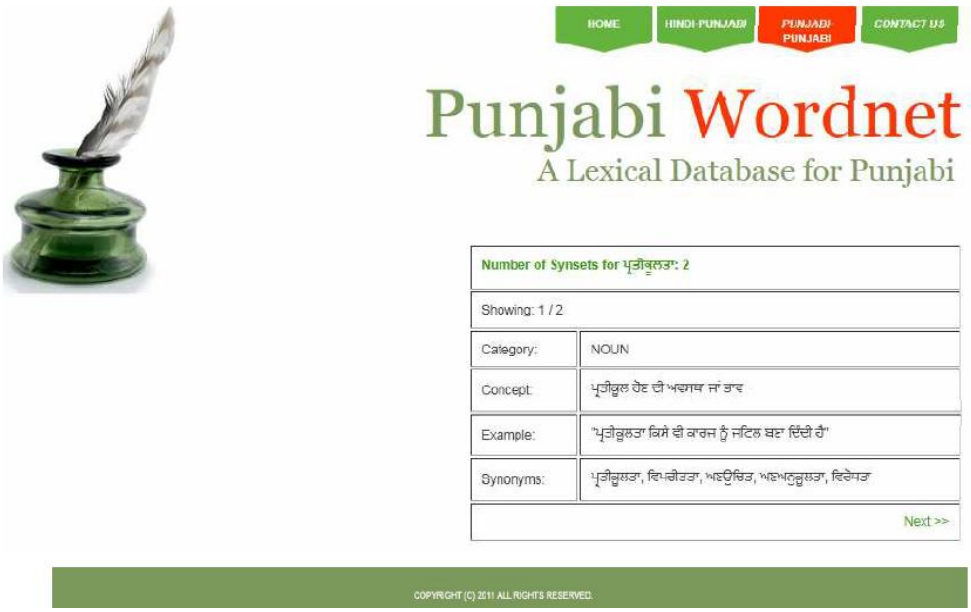

Figure 2.12: Interface showing first sense of the word ਪ੍ਰਤੀਕੂਲਤਾ [20]

User can move to the next sense by clicking on the next link. The second sense of the word ਪ੍ਰਤੀਕੂਲਤਾ *pratīkūltā 'incompatible'* is shown in figure 2.13.

Figure 2.13: Interface showing second sense of the word ਪ੍ਰਤੀਕੂਲਤਾ [20]

Hindi to Punjabi dictionary has been created using Punjabi WordNet. Figure 2.14 shows the web interface of the Hindi to Punjabi dictionary.

Figure 2.14: Web interface for Hindi to Punjabi dictionary [20]

For example, if the user types the word आम *aam 'mango'* from the keypad and click on submit button. It shows the category, concept, one example usage and synonymous words for each of the sense in Punjabi language as shown in figure 2.15.

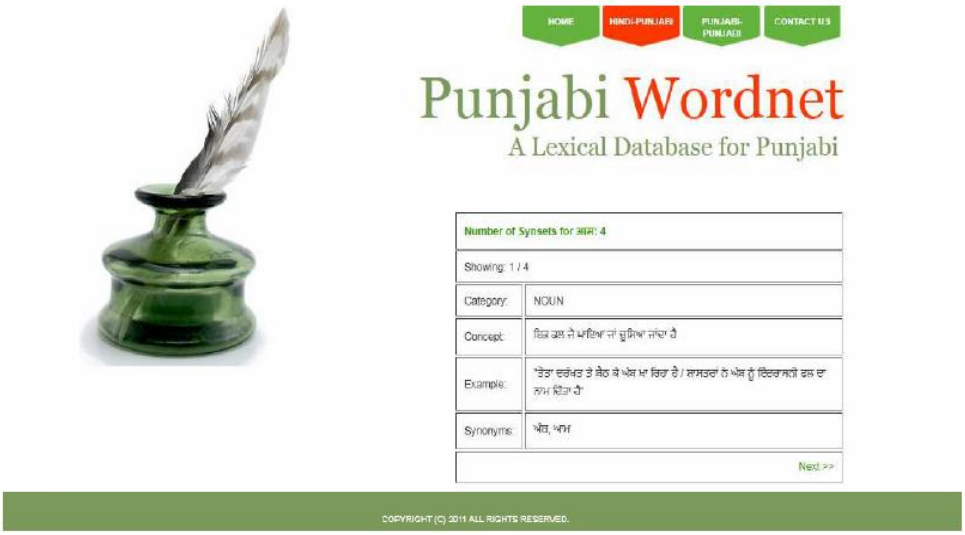

Figure 2.15: Interface showing first sense of the word आम in Punjabi [20]

Punjabi to Hindi dictionary has been created using Punjabi WordNet. Figure 2.16 shows the GUI interface of Punjabi to Hindi dictionary.

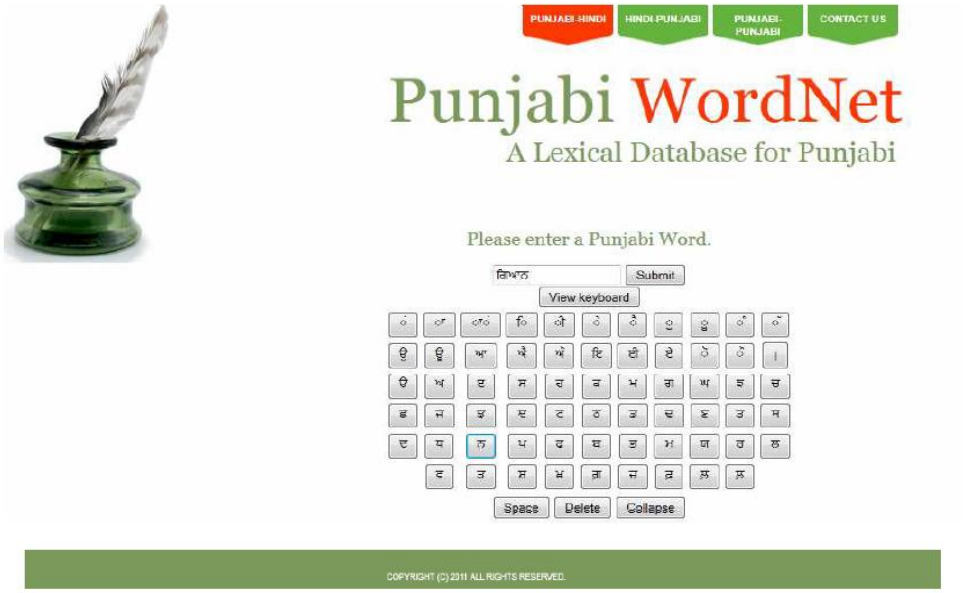

Figure 2.16: Web interface for Punjabi to Hindi dictionary [20]

For example, if the user types the word ਗਿਆਨ *giān 'knowledge'* from the keypad then it shows different senses in which it can be used in Hindi. Figure 2.17 shows the first sense of the word ਗਿਆਨ *giān 'knowledge'*. User can move to next sense by clicking on next button.

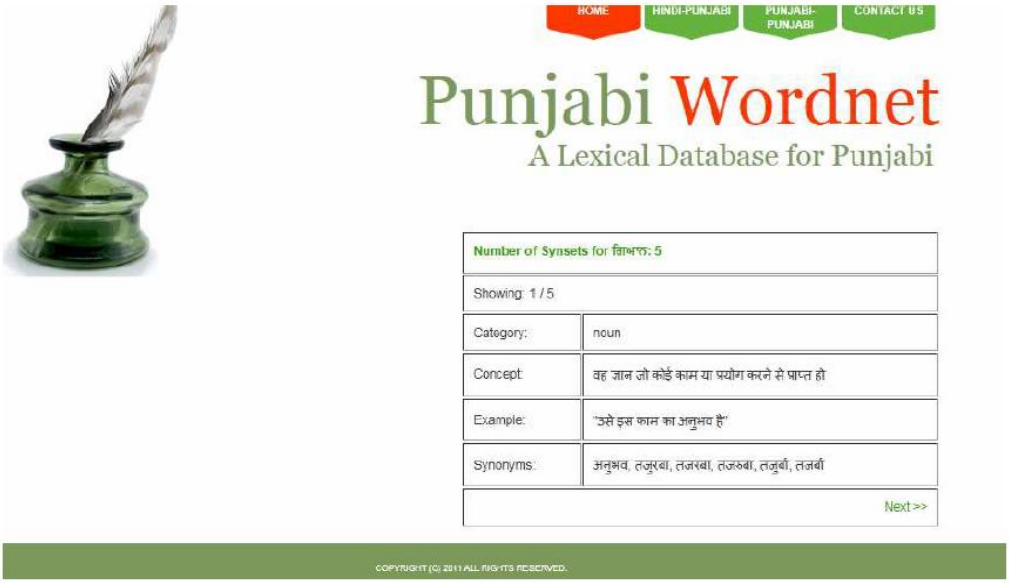

Figure 2.17: Interface showing first sense of the word ਗਿਆਨ in Hindi [20]

The current Punjabi WordNet does not specify lexical and semantic relations. So, the aim of this thesis is to develop lexical and semantic relations for Punjabi WordNet.

Chapter 3
Problem Statement

Most of the countries in the world are using computer and Internet in their own languages. But Indian users are deprived to use computer and Internet resources in their own languages, despite dominance of Indian engineers and scientists in the field of information technology throughout the world. The society at large can benefit from the Information Technology effectively, if people can communicate with computers in their own languages. Thus, there is need to support our languages on technical front to uplift and improve socio-economic condition of our country.

Punjabi language is world's 12^{th} most widely spoken language. It is used in both parts of Punjab, in India and also in Pakistan. Punjabi is the language used by hundreds of millions of people in India, and is also the language used by Punjabis around the world. Surprisingly, little has been done in the field of computerization and lexical resources of this language. It is therefore motivating to develop a Punjabi WordNet. There is a great need to create large scale online lexical-semantic nets for Punjabi language so that Punjabi WordNet can be used for various natural language processing tasks such as information Retrieval, word sense disambiguation, machine translation and other areas. Punjabi WordNet is being developed using expansion approach from Hindi WordNet as both Hindi and Punjabi belong to same language family. Using expansion approach Punjabi WordNet development process become well guided and can save enormous amount of time. At present, Punjabi synsets are being created with reference to Hindi synsets but, the most important part of the WordNet is to link the synsets through lexical and semantic relations. Using expansion approach semantic relations can be easily borrowed from Hindi WordNet as they are same for all the languages. But lexical relations are language specific. Therefore, lexical relations cannot be borrowed from source language. So, there is a need to develop a tool to create the lexical relation entries for the Punjabi WordNet. The main aim of this thesis is to develop lexical creation tool, implement semantic relations for Punjabi WordNet and a web application for Punjabi WordNet so that these resources can be accessed online.

3.1 Objectives

In this thesis, the following objectives have been defined.

1. To create the Punjabi WordNet from Hindi WordNet using expansion approach.
2. To design a database for the storage of synsets and to provide storage for lexical and semantic relations.
3. To port text files containing the synset information to designed database.
4. To develop a lexical creation tool for creation of lexical relations like antonym, compounding, conjunction and gradation *etc*. from Hindi language to any target language.
5. To develop a lexical creation tool for creation of lexical relations like antonym, compounding, conjunction and gradation *etc*. for the words which are not present in Hindi language.
6. To implement semantic relations like hypernymy, hyponymy, meronymy, holonymy, troponymy, entailment and casuation *etc* for Punjabi WordNet.
7. To develop a Web application for Punjabi WordNet for designed database with provision to specify synset and semantic information about a word.

3.2 Methodology

To achieve the above defined objectives, a step-by-step methodology has been followed which is described below.

1. Study of existing WordNets.
2. Analysis of different WordNet relations like synonymy, antonymy, hypernymy, hyponymy, meronymy, holonymy, troponymy, entailment, casuation and gradation *etc*. has been done.
3. Synsets for Punjabi WordNet has been created from Hindi WordNet using IL-MultiDic development.
4. Database has been designed in order to provide storage for lexical and semantic relations.
5. Text files output from IL-MultiDic development tool has been ported to designed database.
6. Antonym creation tool has been developed with the use of NetBeans, with provision to create the antonyms which are present in Hindi WordNet and to create the antonyms for those words which are not covered in Hindi WordNet.

Similarly, tools have been developed for other lexical relations like compounding, conjunction and gradation.

7. Web application for Punjabi WordNet has been developed using JSP at front end and MySql as back end to provide complete information about a word in the form of gloss, example sentence, synonym words and category. Web application also has a provision to show semantic relations like hypernymy, hyponymy, meronymy, holonymy, causation, entailment and troponymy *etc.*

Chapter 4
Design and Implementation

The WordNets are based on the design principles of the English WordNet developed at Princeton University while, paying particular attention to language specific phenomena. The Hindi WordNet was developed by the researchers in the Centre for Indian Language Technology, IIT Bombay. Hindi WordNet has been created manually by looking up the various listed meanings of words in different dictionaries. As Hindi and Punjabi are close members of the same language family. It is convenient to develop Punjabi WordNet from Hindi WordNet by following expansion approach. Using the expansion approach, the synsets of Hindi WordNet are modified to the synsets of Punjabi WordNet through addition or deletion of synonyms in the synset. The semantic relations can be transferred directly, thus saving both time and effort. In the development of Punjabi WordNet, following tasks have been performed.

- To create the synsets for Punjabi from Hindi synsets using IL-MultiDic development tool provided by IIT, Bombay.
- To design the database for Punjabi WordNet.
- To port the text files output from IL-MultiDic development tool to the designed database.
- To design algorithm for extraction of synset information like gloss, example, category and synonym words.
- To design algorithm for creation of lexical relation entries like antonymy, gradation, conjunction and compounding.
- To design algorithm for retrieving semantic relations like hypernymy, hyponymy, meronymy, holonymy, troponymy and entailment for Punjabi WordNet.

4.1 Creation of synsets using IL-MultiDic development tool

For creation of synsets in Punjabi language, IL-MultiDic development tool has been used. The tool is designed to maintain standardization of the lexical data entries by using the same ***synset_id*** numbers to represent similar concepts in source and target languages. The data values are to be generated for Punjabi in correspondence to the entries done for Hindi language. It helps in standardizing the lexical entries to be used.

4.1.1 Configuration of IL-MultiDic development tool

While configuring the tool, source and target files needs to be selected. The source and target files act as input and output to the tool. The source file consists of synsets which have already been created in Hindi Language and which guides the synset creation for target language. The target file consists of synsets which is created using this tool. Browse button is used to choose the source and target files. Target language specifies the language of the target file. In this case, target language is Punjabi and Target font is Arial Unicode MS as shown in Figure 4.1.

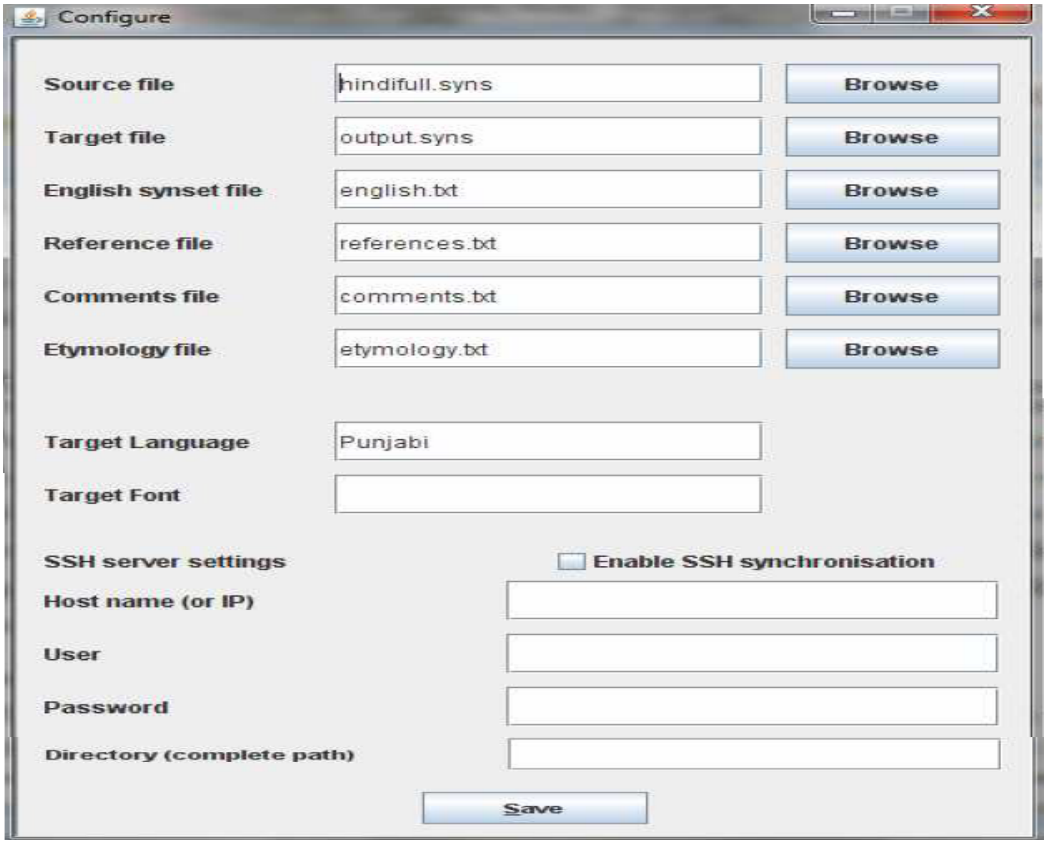

Figure 4.1: Configuring IL-MultiDic development tool

4.1.2 Working of IL-MultiDic development tool

This tool is a user friendly and it is used to link together the synsets in different languages that convey the similar meaning. In this case, the tool is being used to link the synsets for Hindi and Punjabi language. First of all source and target files are uploaded during configuration of the tool. The files which contain the data entries for Hindi words are the source files. Each synset has been assigned a unique id number which uniquely recognize the concept represented by that synset.

Figure 4.2 shows the tool used for the creation of Punjabi synsets corresponding to each Hindi word entry.

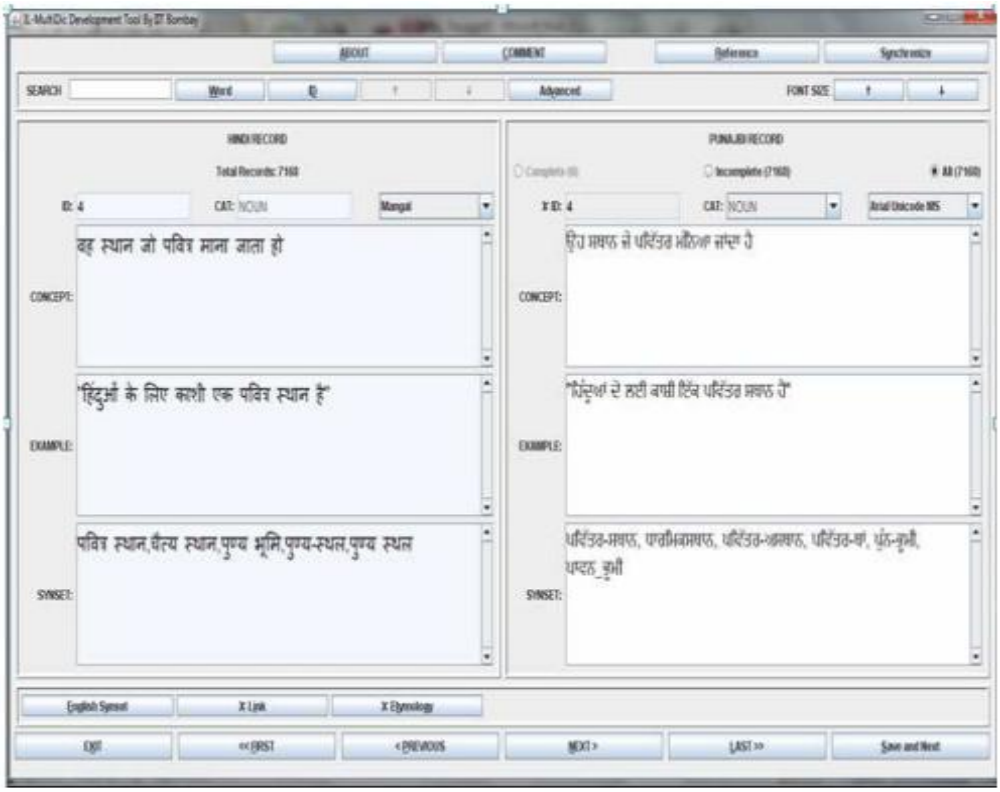

Figure 4.2: Tool for creating standardized lexical data

The data entries of the files used by Hindi WordNet are shown on the left hand side for which corresponding entries for the Punjabi language has to be done on the right side as shown in figure 4.2. Each source file contains a list of synsets for all parts of speech, *i.e.*, noun, verb, adjective and adverb. Each of the record contains ***synset_id*** number, synonymous word forms, category, concept and an example sentence.

4.2 Database design for Punjabi WordNet

The output of IL-MultiDic development tool is stored in text file. The text file consists of synset information only and there is no provision to store lexical and semantic relations. In order to provide storage for lexical and semantic relations, database has been designed. The database has been divided in two parts referring to the database design proposed by Indradhunsh WordNet team [7].

- WordNet_Punjabi for storage of synset information
- WordNet_Master for storage of semantic information

4.2.1 Database design for storage of synset information

Synset information is not same for all the languages. This database contains the tables having information related to the target language. It includes all the tables that maintain the information regarding synset details, words in the language, examples and all the lexical relations like antonymy, gradation, compounding and conjunction *etc*. The description of all the tables representing synset information is given below.

1) **Table Name:** wn_synset

   Purpose: wn_synset _table has been used to store the synset information in the form of ***concept_definition***, ***category_id*** and ***source_id***.

Table 4.1: Description of wn_synset

Sr. No.	Field Name	Purpose
1	synset_id	Primary key: Uniquely identifies a concept/synset in the language
2	concept_definition	The gloss / concept definition in a synset
3	category_id	Foreign key from category table. Specifying if the concept is a noun, verb, adjective or adverb
4	source_id	Foreign key from source table. Specifies the source from where the concept is taken

2) **Table Name:** wn_word

   Purpose: wn_word table has been used to store unique words of the language. It holds the vocabulary of the language.

Table 4.2: Description of wn_word

Sr. No.	Field Name	Purpose
1	word_id	Primary key: Uniquely identifies a word of the language.
2	Word	The word of the language.

3) **Table Name:** wn_synset_words

   Purpose: wn_synset_word table is used to store synonymous words in a synset which are used to describe a concept in a language.

Table 4.3: Description of wn_synset_words

Sr. No.	Field Name	Purpose
1	synset_id	Foreign key from wn_synset table.
2	word_id	Foreign key from wn_word table.
3	word_priority	Gives the priority for the synonymous words.

4) Table Name: wn_synset_example

Purpose: wn_synset_example table has been used to store example sentences for a concept / synset.

Table 4.4: Description of wn_synset_example

Sr. No.	Field Name	Purpose
1	synset_id	Foreign key from wn_synset table.
2	example_content	The example sentence text.
3	example_priority	Gives the priority for the example sentences. The order in which the examples are to be displayed.

5) Table Name: wn_rel_antonymy

Purpose: wn_rel_antonymy is used to store antonym relation between a pair of words.

Table 4.5: Description of wn_rel_antonymy

Sr. No.	Field Name	Purpose
1	word_id	Foreign key from the word table. Points to the word for which the antonym is being set.
2	synset_id	Foreign key from the synset table. Points to the synset corresponding to the word.
3	anto_word_id	Foreign key from the word table. Points to the antonym word.
4	anto_synset_id	Foreign key from the synset table. Points to the synset corresponding to the anto word.
5	anto_grad_property_id	Foreign key from wn_property_anto_grad table. Points to the property name.

6) Table Name: wn_rel_gradation

Purpose: wn_rel_gradation has been used to store the gradation relation between three words.

Table 4.6: Description of wn_rel_gradation

Sr. No.	Field Name	Purpose
1	first_word_id	Foreign key from the word table. Points to the first word for which the gradation is being set.
2	first_synset_id	Foreign key from the synset table. Points to the synset corresponding to the word sense for the first word
3	mid_word_id	Foreign key from the word table. Points to the mid word for which the gradation is being set.
4	mid_synset_id	Foreign key from the synset table. Points to the synset corresponding to the word sense for the mid word
5	last_word_id	Foreign key from the word table. Points to the last word for which the gradation is being set.
6	last_synset_id	Foreign key from the synset table. Points to the synset corresponding to the word sense for the last word
7	anto_grad_property_id	Foreign key from wn_property_antonymy_gradation table. Points to the property name based on which the antonym is chosen. Example colour, time, gender etc

7) Table Name: wn_rel_compounding

Purpose: wn_rel_compounding table is used to maintain the compound words of the language.

Table 4.7: Description of wn_rel_compounding

Sr. No.	Field Name	Purpose
1	compound_word_id	Foreign key from the word table. Points to a

		compound word which is formed of two or more words.
2	compound_synset_id	Foreign key from the synset table. Points to the synset corresponding to the word compound word
3	part_word_id	Foreign key from the word table. Points to a part of the compound word.
4	part_synset_id	Foreign key from the synset table. Points to the synset corresponding to the part of the compound word

8) Table Name: wn_rel_conjunction

Purpose: wn_rel_conjunction table maintains the words formed by conjunction of words in the language.

Table 4.8: Description of wn_rel_conjunction

Sr. No.	Field Name	Purpose
1	conjunction word_id	Foreign key from the word table. Points to a conjunction word which is formed of two or more words.
2	conjunction_synset_id	Foreign key from the synset table. Points to the synset corresponding to the conjunction word.
3.	part_word_id	Foreign key from the word table. Points to a part of conjunction word.
4.	part_synset_id	Foreign key from the synset table. Points to the synset corresponding to the part word.

4.2.2 DataBase design for storage of semantic information

Semantic information is shared by all the languages. This database maintains tables which borrow the relations from the source WordNet (Hindi WordNet). It includes all the tables that keep the semantic information about a synset like hypernymy, hyponymy, meronymy, holonymy, troponymy and enatilament *etc*. The description of all the tables representing the semantic information is given below.

9) **Table Name:** wn_rel_hypernymy_hyponymy

Purpose: wn_rel_hypernymy_hyponymy has been used to store hypernymy and hyponymy type of a relation which is a IS-A-KIND-OF type of a semantic relationship between synsets. For example, rose is a kind of flower then rose is child or hyponymy and flower is parent or hypernymy.

Table 4.9: Description of wn_rel_hypernymy_hyponymy

Sr. No.	Field Name	Purpose
1	parent_synset_id	Foreign key from the synset table. Points to the parent concept which is called hypernymy of the IS-A-KIND-OF relationship.
2	child_synset_id	Foreign key from the synset table. Points to the child concept which is called hyponymy of the IS-A-KIND-OF relationship.

10) **Table Name:** wn_rel_meronymy_holonymy

Purpose: wn_rel_meronymy_holonymy table maintains meronymy and holonymy type of a relation. This is a PART-WHOLE type of a semantic relationship between synsets. For example, leaf is part of tree here tree is whole or meronym and leaf is part or holonym.

Table 4.10: Descriptions of wn_rel_meronymy_holonymy

Sr. No.	Field Name	Purpose
1	whole_synset_id	Foreign key from the synset table. Points to the whole concept that is meronym of the PART-WHOLE relationship.
2	part_synset_id	Foreign key from the synset table. Points to the part concept that is holonymy of the PART-WHOLE relationship.
3	mero_holo_property_id	Foreign key from the property_meronymy_holonymy table. Points to the additional description about the relation.

11) Table Name: wn_rel_troponymy

Purpose: wn_rel_troponymy has been used to store the troponymy type of a semantic relationship between synsets.

Table 4.11: Description of wn_rel_troponymy

Sr. No.	Field Name	Purpose
1	synset_id	Foreign key from the synset table.
2	troponym_synset_id	Foreign key from the synset table. Points to the troponym synset.

12) Table Name: wn_rel_entailment

Purpose: wn_rel_entailment table maintains entailment type of a semantic relationship between synsets.

Table 4.12: Description of wn_rel_enatilment

Sr. No.	Field Name	Purpose
1	synset_id	Foreign key from the synset table.
2	entailed_synset_id	Foreign key from the synset table. Points to the entailed synset.

13) Table Name: wn_rel_causative

Purpose: wn_rel_causative table maintains the causative semantic relation between synsets.

Table 4.13: Description of wn_rel_causative

Sr. No.	Field Name	Purpose
1	synset_id	Foreign key from the synset table. Points to a synset/concept.
2	causes_synset_id	Foreign key from the synset table. Points to a cause synset/concept.

14) Table Name: wn_property_meronymy_holonymy

Purpose: wn_property_meronymy_holonymy table maintains the different types of relation properties for relations like meronymy and holonymy have properties like component-object, feature-activity *etc*.

Table 4.14: Description of wn_property_meronymy_holonymy

Sr. No.	Field Name	Purpose
1	mero_holo_property_id	Primary key: Uniquely identifies a property type.
2	mero_holo_property_value	The name of the property like component-object, feature-activity etc

15) Table Name: wn_relation_types

Purpose: wn_relation_types table has been used to store the relation information of all the relation tables in the form of ***relation_id***, ***rel_description*** and ***table_name***.

Table 4.15: Description of wn_relation_types

Sr. No.	Field Name	Purpose
1	relation_id	Foreign Key from relation types table. Point to a relation in which the synset belongs
2	rel_description	Label to display the name of the relation
3	table_name	Name of the table where relations are stored

16) Table Name: wn_semantic_relations

Purpose: wn_semantic_relations table maintains the semantic relations with respect to the synsets.

Table 4.16: Description of wn_semantic_relations

Sr. No.	Field Name	Purpose
1	synset_id	Foreign Key from synset table. Points to a synset.

2	relation_id	Foreign Key from relation types table from WordNet_Master database. Point to a relation to which the synset belongs.

To provide storage for synset and semantic information, database has been described in this section. The next step is to populate the database with the records stored in text files, output from IL-Multidic development tool. The porting of text files to the designed database is further described in next section.

4.3 Porting of text files to MySql database

The text files created using IL-MultiDic development tool are ported to MySql database using DB Import tool. The Punjabi synset file is given as input to tool and it automatically creates and populate WordNet_Master and WordNet_Punjabi databases for the Punjabi language. The database created using this tool is used as backend for creation of Punjabi WordNet.

Figure 4.3 depicts the interface for DB Import tool. In order to create the WordNet_Master database click on create and populate master database button.

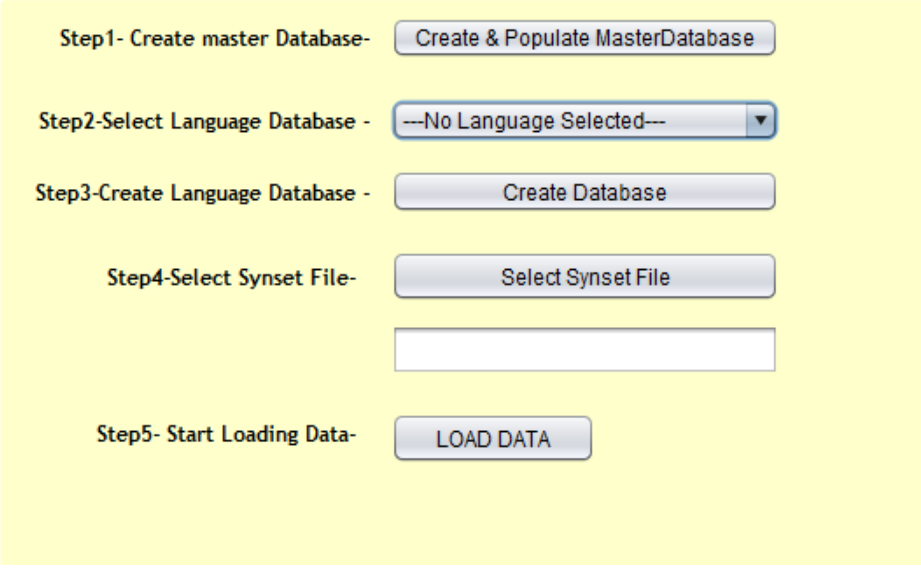

Figure 4.3: DB Import Tool

To create WordNet_Punjabi database, select target language and click on create database button. The next step is to populate the database with target language

records. This can be done by selecting the synset file for the target language and click on load data button.

4.4 Extraction of complete information about a word

The complete information of a word comprises of concept definition, example content, synonym words and category information, *i.e.*, whether it is noun, verb, adjective or adverb. All this information is stored in different database tables as discussed in section 4.2. Therefore, we need to join all the tables in order to extract whole information. Algorithm has been designed to extract the information about a word is given below.

Algorithm 4.1: To extract full information about a word

1. Extract the ***word_id*** of the input word from ***wn_word*** table.
2. Extract the ***synset_ids*** from the ***wn_synset_word*** table for the ***word_id*** found in step 1.
3. For each of the ***synset_id*** found in step 2 extract the corresponding ***concept_definition*** and ***category_id*** from the ***wn_synset*** table.
4. Extract the corresponding ***category_value*** from ***wn_master_category*** for the ***category_id*** found in step 3.
5. Extract the corresponding ***example_content*** from ***wn_synset_example*** for each of the ***synset_ids*** found in step 2.
6. For each ***synset_id*** found in step 2, extract the ***word_ids*** from the ***wn_synset_word*** table.
7. Extract the corresponding words from the ***wn_word*** table for each ***word_id*** found in step 6.

Description of Algorithm 4.1

Let us consider the word ਆਗਿਆ *āgiā 'permission'* for the explanation of algorithm 4.1. First of all, the system searches the particular word ਆਗਿਆ in the ***wn_word*** table and extracts the ***word_id*** for the particular word. In our example, the ***word_id*** retrieved for the word ਆਗਿਆ *āgiā 'permission'* is 12220 as shown in step1 of figure 4.4. The ***word_id*** found is then used to find the ***synset_ids*** from ***wn_synset_word*** table. The ***synset_id*** found for the word ਆਗਿਆ is 105 as referred in step 2 of figure 4.4. For a given ***synset_id*** 105, the system refers to ***wn_synset_table*** to extract the concept and category information as shown in step 3 of figure 4.4. The ***category_id***

found is then mapped to ***wn_master_category*** table to find the details regarding whether the word in noun, verb, adjective and adverb. For the word ਆਗਿਆ *āgiā 'permission'* ***category_id*** is found to be 1 which is mapped to noun in ***wn_master_category*** table as illustrated in step 4.

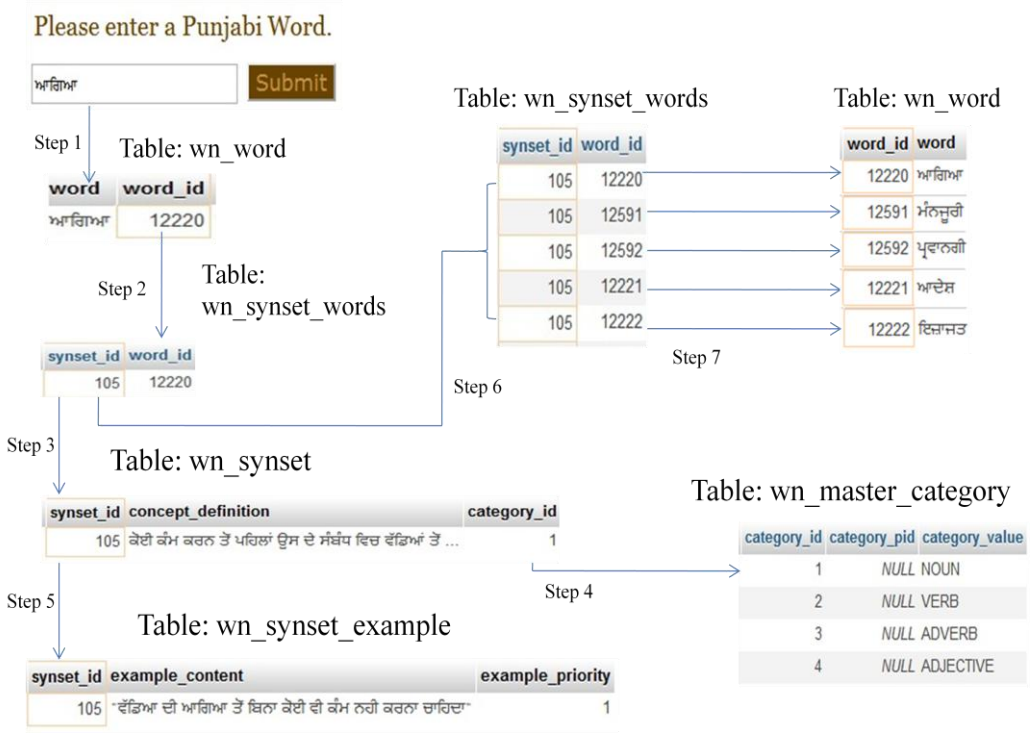

Figure 4.4: Extracting complete information for word ਆਗਿਆ

In order to display the example for the particular word s ***wn_synset_example*** table has been referred. The system finds the given ***synset_id*** in the ***wn_synset_example*** table and retrieves the corresponding example content as shown in step 5 of figure 4.4. To find out the synonym words for a given ***synset_id*** the system refers to ***wn_sysnset_word*** table and extracts the corresponding ***word_ids*** for a given ***synset_id*** as shown in step 6 of the figure 4.4. Then from the ***word_ids*** corresponding words are extracted from ***wn_word*** table as sown in step 7 of figure 4.4. To present the information retrieved using this approach a web interface has been designed. The complete information for the word ਆਗਿਆ *āgiā 'permission'* is displayed in web interface as shown in figure 4.5.

Figure 4.5: Interface showing synset information for the word ਆਗਿਆ

4.5 Lexical creation tool

Using the expansion approach semantic relations are borrowed from the source language as they are same for all the languages. But, lexical relations are language specific, so they cannot be borrowed from the source language using expansion approach. In order to create the lexical relations for the Punjabi language, one can do the entries manually, but it is a time consuming task. The better approach is to extend the relation from Hindi WordNet to Punjabi WordNet. But extending the lexical relation from Hindi WordNet does not solve the purpose, because lexical relations may exists between the words which are not present in Hindi language but present in the target language. It is convenient to design a lexical creation tool with a provision to create the lexical relations from Hindi WordNet and for the words which are not present in Hindi WordNet. Lexical creation tool has been designed for the following lexical relations.

- Antonym
- Compounding
- Conjunction
- Gradation

4.5.1 Creation of antonyms with reference to Hindi WordNet

The antonyms for the Hindi WordNet have already been created. Antonyms for the Punjabi WordNet can be created from the antonyms of Hindi WordNet, but database design of Punjabi WordNet is different from Hindi WordNet. An efficient approach needs to be followed which can bridge the gap between two different database designs and create the antonyms for the Punjabi WordNet from Hindi WordNet. The algortihm for creation of antonyms from Hindi WordNet is given below.

Algorithm 4.2: To create antonyms with reference to Hindi WordNet

1. Extract the ***synset_id*** of the ***synset_word*** from the ***tbl_noun_anto_direction*** table.
2. Extract the ***word_ids*** from ***wn_synset_word*** table, for the ***synset_id*** found in step 1.
3. For each ***word_id*** found in step 2, extract the corresponding words from ***wn_word*** table.
4. Extract the ***synset_id*** of the antonym ***synset_word*** from the ***tbl_noun_anto_direction*** table.
5. Extract the ***word_ids*** from ***wn_synset_word*** table, for the ***synset_id*** found in step 4.
6. For each ***word_id*** found in step 5, extract the corresponding words from ***wn_word*** table.

Description of Algorithm 4.2

For example, for the word पूर्व *purav 'east'* in Hindi, the system searches for the word in the ***tbl_noun_anto_direction*** table and extract corresponding ***synset_id*** which is found to be 6898 as shown in figure 4.6.

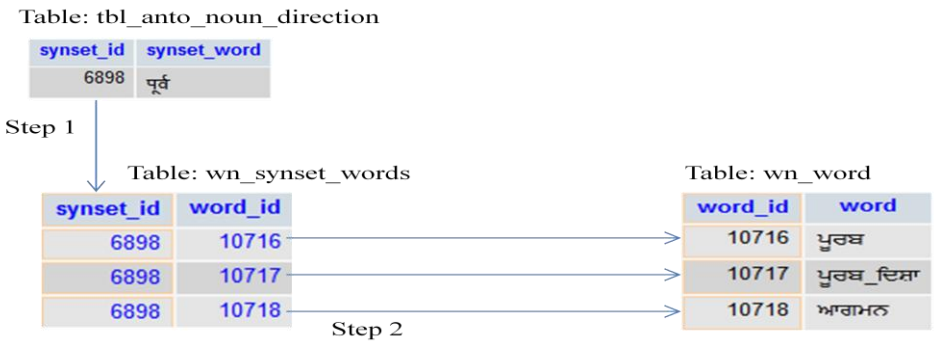

Figure 4.6: Extracting words corresponding to ***synset_id*** 6898

For the given ***synset_id*** 6898, system refers to ***wn_synset_word*** table to extract the ***word_ids*** as shown in step 1 of figure 4.6. For each of the ***word_id*** found, system retrieves the corresponding words from the ***wn_word*** table as illustrated in step 2 figure 4.5.

The similar approach is being followed to find the antonym words for the antonym ***synset_id.*** The extraction of antonyms words corresponding to ***synset_id*** 6616 for Punjabi language for the as shown in figure 4.7.

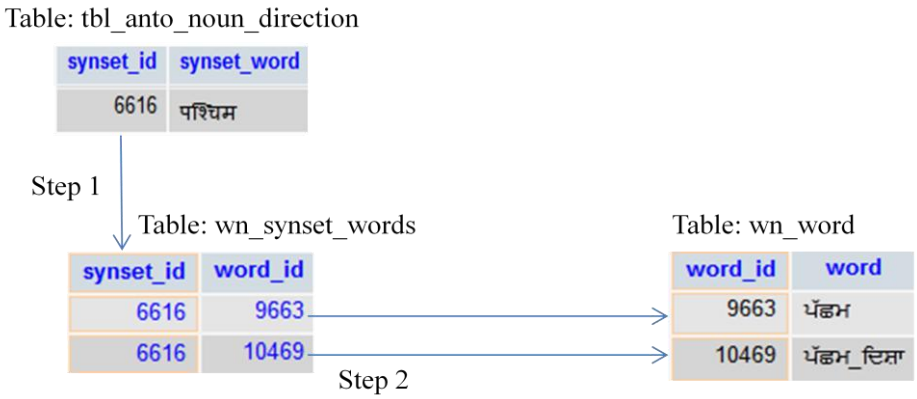

Figure 4.7: Extracting words corresponding to antonym ***synset_id*** 6616

To display the information retrieved using the algorithm 4.2 an interface has been designed. For example, the information corresponding to ***synset_id*** 6898 and ***anto_synset_id*** 6616 has been shown in figure 4.8.

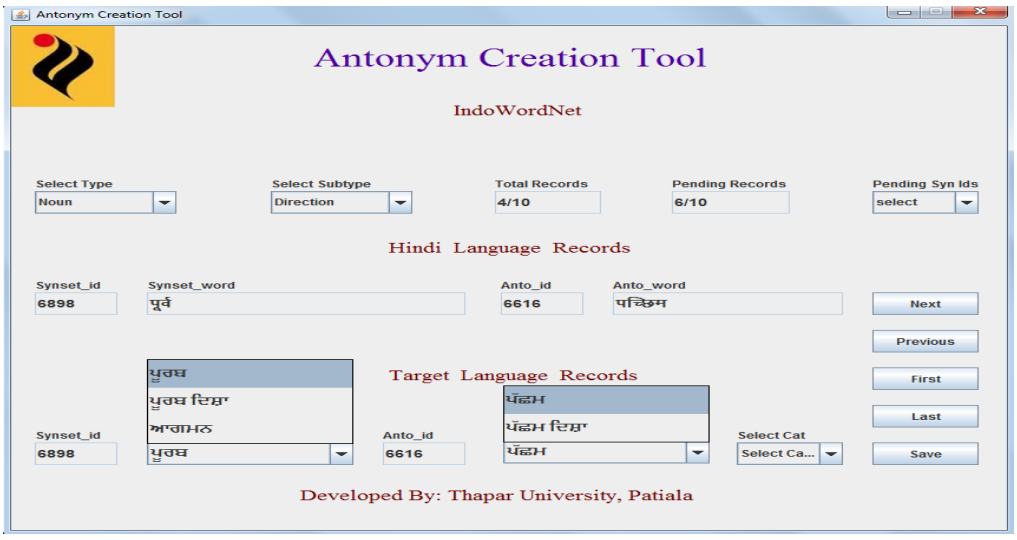

Figure 4.8: Interface showing synset and antonym synset words

4.5.2 Creation of antonyms without reference to Hindi WordNet

The antonym relation may also exist between the words which are not present in Hindi WordNet, but present in the target language. There is need to design a tool which can create the antonyms for such words. The algorithm followed in creation of such tool is given below.

Algorithm 4.3: To create antonyms without reference to Hindi WordNet

1. Extract the ***word_id*** of the input from the ***wn_word*** table.
2. Extract the ***synset_ids*** from ***wn_synset_word*** table, for the ***word_id*** found in step 1.
3. For each ***synset_id*** found in step 2, extract the corresponding concepts from ***wn_synset*** table.
4. Extract the ***word_id*** of the input antonym word from the ***wn_word*** table.
5. Extract the ***synset_ids*** from ***wn_synset_word*** table, for the ***word_id*** found in step 4.
6. For each ***synset_id*** found in step 5, extract the corresponding concepts from ***wn_synset*** table.

Description of Algorithm 4.3

Consider the input word ਚੰਗਾ *caṅgā ‘good character’*, the system refers to ***wn_word*** table to search the input word and extracts the corresponding ***word_id***. The ***word_id*** for the input word ਚੰਗਾ *caṅgā ‘good character’* is 166 as shown in step 1 of figure 4.9.

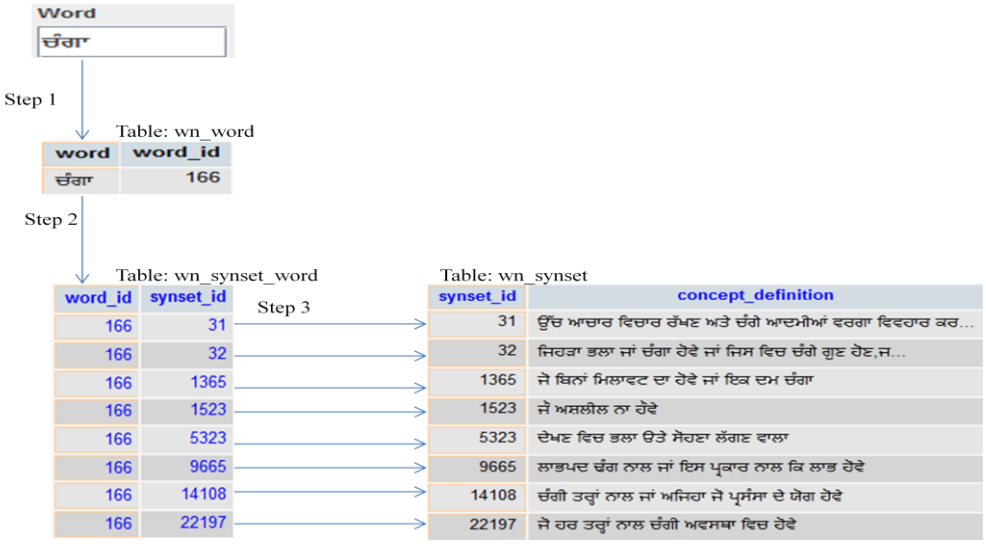

Figure 4.9: Extraction of concepts for the word ਚੰਗਾ

For the given ***word_id*** 166, system searches for the different ***synset_ids*** in the ***wn_synset_word*** table as shown in step 2 figure 4.9. In order to retrieve the concepts for the input word ਚੰਗਾ system refers to ***wn_synset*** table and extracts the corresponding concepts for each of the ***synset_id*** found in step 2 as shown step 3 of figure 4.9.

The similar approach can be followed to find the concepts for the antonym word. The concepts for the word ਭੈੜਾ *bhaiṛā ‘characterless’* can be found as shown in figure 4.10

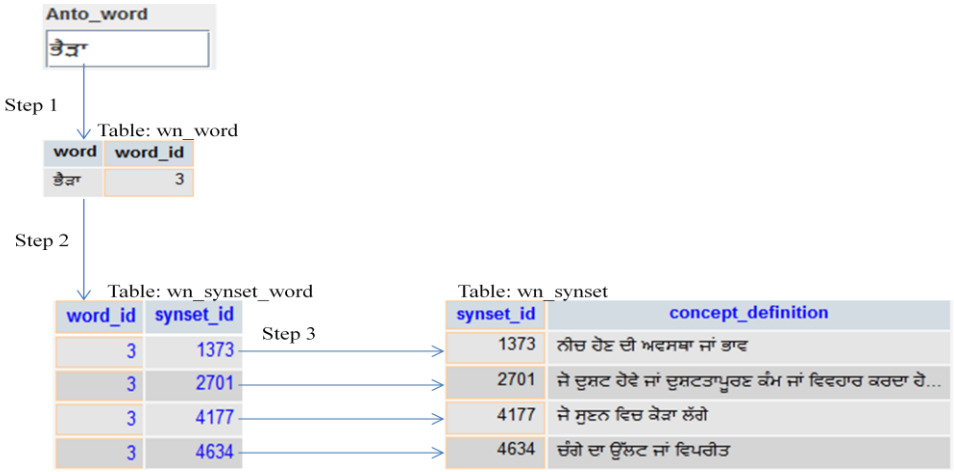

Figure 4.10: Extraction of concepts for the word ਭੈੜਾ

To present the information retrieved using the algorithm 4.3 an interface has been designed. The information corresponding to the word ਚੰਗਾ *caṅgā ‘good* character’ and antonym word ਭੈੜਾ *bhaiṛā ‘characterless’* is shown in figure 4.11.

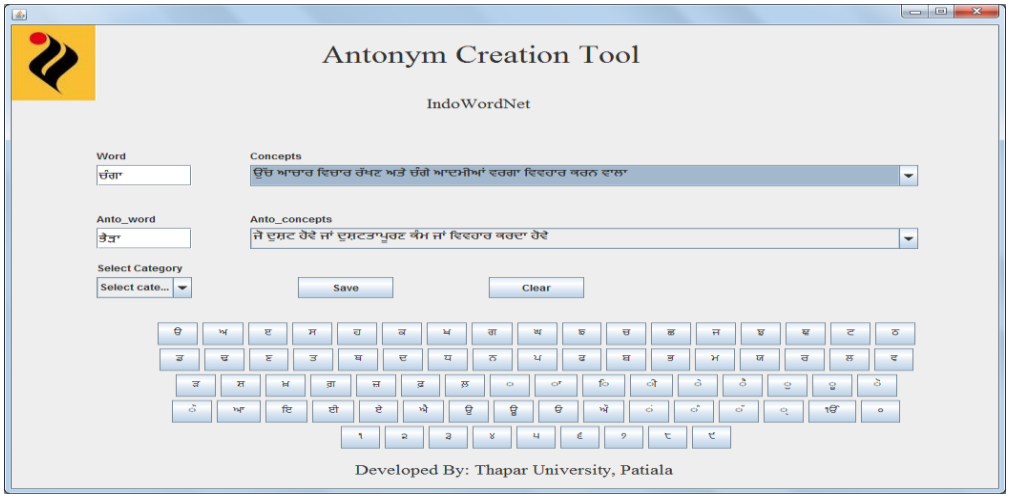

Figure 4.11: Interface for creation of antonyms without Hindi WordNet

The approach followed creation of antonym creation tool can also be applied to create the tools for compounding, conjunction and gradation *etc.*

4.6 Extraction of semantic relations for a word

The various semantic relations are hypernymy, hyponymy, meronymy, holonymy, casuation, troponymy and entailment *etc*. All the semantic relations are stored in different database tables according to the database design discussed in section 4.2. In order to retrieve semantic relations of a word, we need to join tables containing semantic information. The algorithm to extract the semantic information about a word is given below.

Algorithm 4.4: To extract semantic relations for a word

1. Extract the ***word_id*** of the input word from the ***wn_word*** table.
2. Extract the ***synset_ids*** from the ***wn_synset_word*** table for the ***word_id*** found in step 1.
3. Extract the corresponding ***relation_ids*** from ***wn_semantic_relations*** table for each of the ***synset_id*** found in step 2.
4. For each of the ***relation_id*** found in step 3, extract the corresponding ***relation_description, relation_table*** and ***table_from_column1*** from ***wn_relation_types*** table.
5. Extract the corresponding ***synset_ids*** from the ***table_name*** found in step 4 corresponding to the ***synset_id*** found in step 2.
6. For each of the ***synset_id*** found in step 5, extract the corresponding ***concept_definition*** and ***category_id*** from ***wn_synset*** table.
7. For each of the ***category_id*** found in step 6, extract the ***category_value*** from ***wn_master_category***.
8. For each of the ***synset_id*** found in step 5, extract the corresponding ***example_content*** from ***wn_synset_example.***
9. Extract the corresponding ***word_ids*** from the ***wn_synset_word*** table for each ***synset_id*** found step 5.
10. For each ***word_id*** found in step 9, extract the corresponding words from ***wn_word*** table.

Description of Algorithm 4.4

For example, user enters the word ਆਗਿਆ *āgiā 'permission'* to extract its complete synset and semantic information. The system searches the particular word ਆਗਿਆ *āgiā 'permission'* in the ***wn_word*** table and extracts the ***word_id*** for the particular word. In our example, the ***word_id*** extracted for the word ਆਗਿਆ *āgiā 'permission'* is 12220 as shown in step1 of figure 4.12.

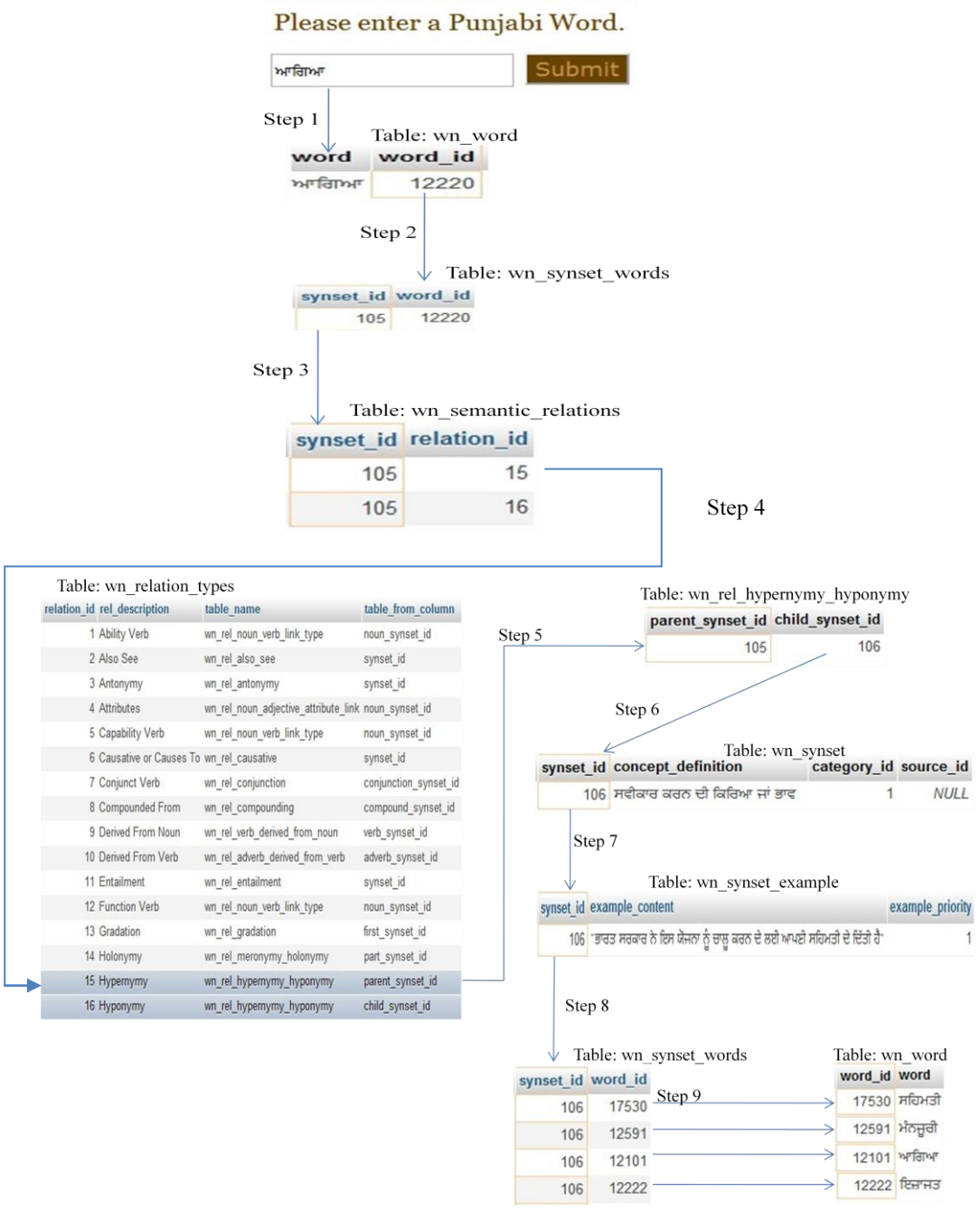

Figure 4.12: Extracting semantic relations for word ਆਗਿਆ

The ***word_id*** found is then used to find the ***synset_ids*** from ***wn_synset_word*** table. The ***synset_id*** found for the word ਆਗਿਆ *āgiā 'permission'*is 105 as shown in step 2 figure 4.12. In order to find out the semantic relations for a given ***synset_id***, system refers to ***wn_semantic_relations***. The synset may be involved in more than one semantic relation. For example, corresponding to ***synset_id*** 105, there are two ***relation_ids*** 15 and 16 as illustrated in step 3 of figure 4.12. The details of these ***relation_ids*** is extracted from ***wn_relation_types*** table which includes relation in which a particular synset is involved and table for extracting the information for the corresponding relation and column involved. The word ਆਗਿਆ *āgiā 'permission'* is involved in hypernymy and hyponymy relation as shown in step 4 of the figure 4.12. In step 5, for each of the extracted relation, corresponding relation table (found in step 4) is referred to find out the ***synset_ids*** which are in corresponding relation (found in step 4) with the given ***synset_id.*** For example, to extract the ***synset_ids*** which are in hypernymy relation with ***synset_id*** 105, the ***child_synset_ids*** are retrieved from the ***wn_rel_hypernymy_hyponymy*** table (found in step 4). The ***child_synset_id*** found to be 106. For the ***synset_id*** 106, system refers to ***wn_synset*** table to display the concept and the category information as shown in step 6 of figure 4.12. In order to find out the example content, system finds the given ***synset_id*** 106 in the ***wn_synset_example*** table and retrieves the corresponding ***example_content*** as shown in step 7 of figure 4.12. To find out the synonym words for a given ***synset_id*** 106 the system refers to ***wn_sysnset_word*** table and extracts the corresponding ***word_ids*** for a given ***synset_id*** 106 as depicted in step 8 figure 4.12. Then from the ***word_ids*** corresponding words are extracted from ***wn_word*** table as shown in step 9 of figure 4.12.

The complete information retrieved using this approach is presented to user in web interface. The hypernymy relation retrieved for the word ਆਗਿਆ *āgiā 'permission'* is shown in figure 4.13.

Figure 4.13: Interface showing hypernymy relations for the word ਆਰਿਆ

In this chapter, processes of creation of synsets for Punjabi WordNet and database design for storage of synset and semantic information have been described. Algorithms for creation of lexical creation tools and extraction of complete synset and information about a word have been discussed in this chapter. The working of various tools lexical creation and web interface for Punjabi WordNet has been described in next chapter.

Chapter 5
Results and Discussion

In this chapter, the tools developed for creation of lexical relations and web application for implementation of semantic relations has been described. Tools have been developed for the creation of following lexical relations.

- Antonymy
- Gradation
- Compounding
- Conjunction

These tools are further divided into two categories as shown in figure 5.1

- Creation of lexical relations for the words which are present in Hindi WordNet.
- Creation of lexical relations for the words which are not present in Hindi WordNet.

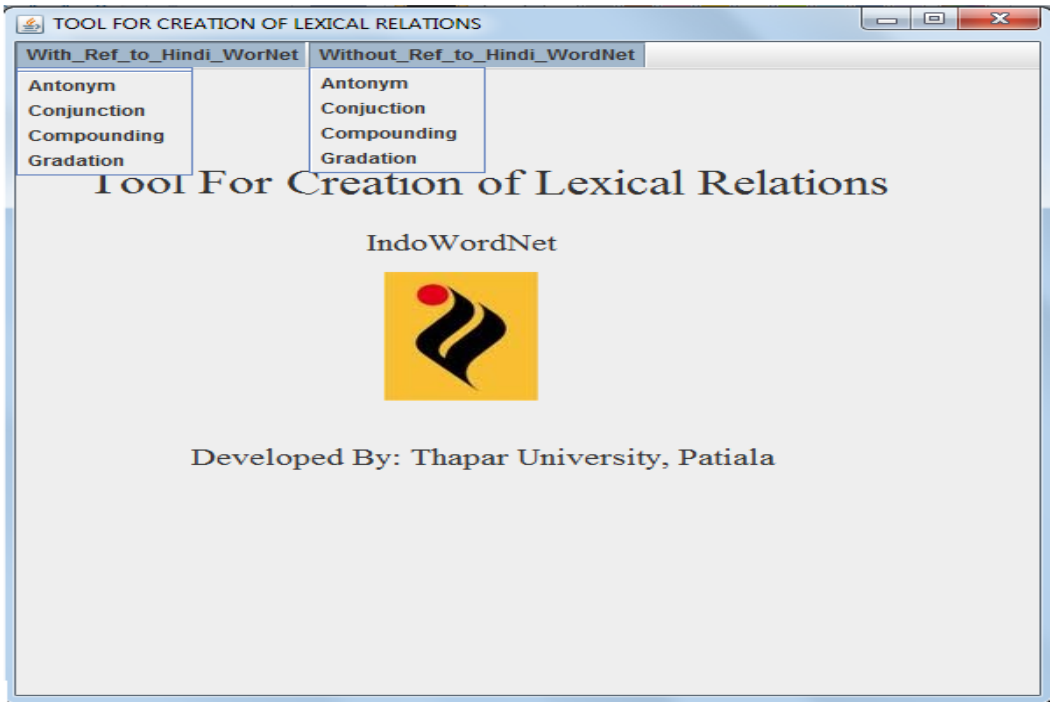

Figure 5.1: Lexical Creation Tool

5.1 Antonymy creation tool with reference to words in Hindi WordNet

Antonym creation tool helps to create the antonyms for any language. The tool is developed keeping in mind to reduce the manual typing effort for the creation of antonyms. The basis of this tool is the antonyms for Hindi language. The tool provides a user friendly interface for creation of antonyms for target language. User needs to select one word from the antonyms of the Hindi language, corresponding to the ***synset Id*** of that word all the words for the target language is displayed. Similarly for the antonym word, all the words corresponding to that ***synset Id*** are displayed. Now user can select the best match for the two words for the creation of antonyms for the target language.

Hindi WordNet contains 48 tables for specifying the antonyms in Hindi language. These tables are classified in four major categories namely noun, verb, adjective, and adverb. All the four major categories are further classified into twelve subcategories action, amount, direction, color, gender, personality, place, time, state, manner, quality and size. So, an interface has been designed to select the major category and sub category as shown in figure 5.2

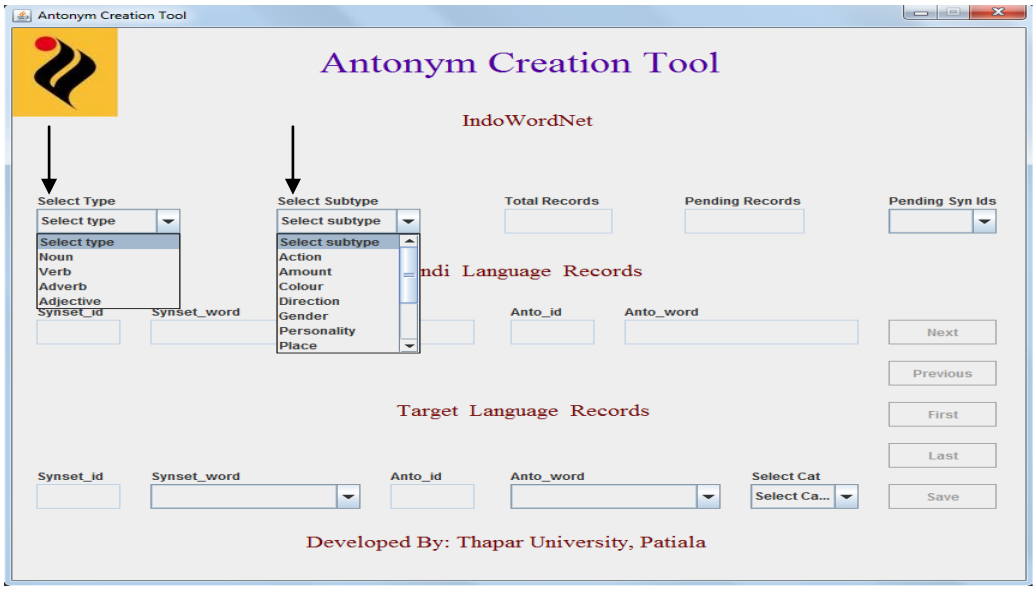

Figure 5.2: Description of type and subtype

Whenever both category and subcategory is selected, a table is selected, total number of records and pending records are shown in corresponding textboxes. It shows current record, number of pending synsets and total number of records corresponding to

currently selected table. Moreover, pending ***synset_Ids*** are also displayed in drop down list as shown in figure 5.3.

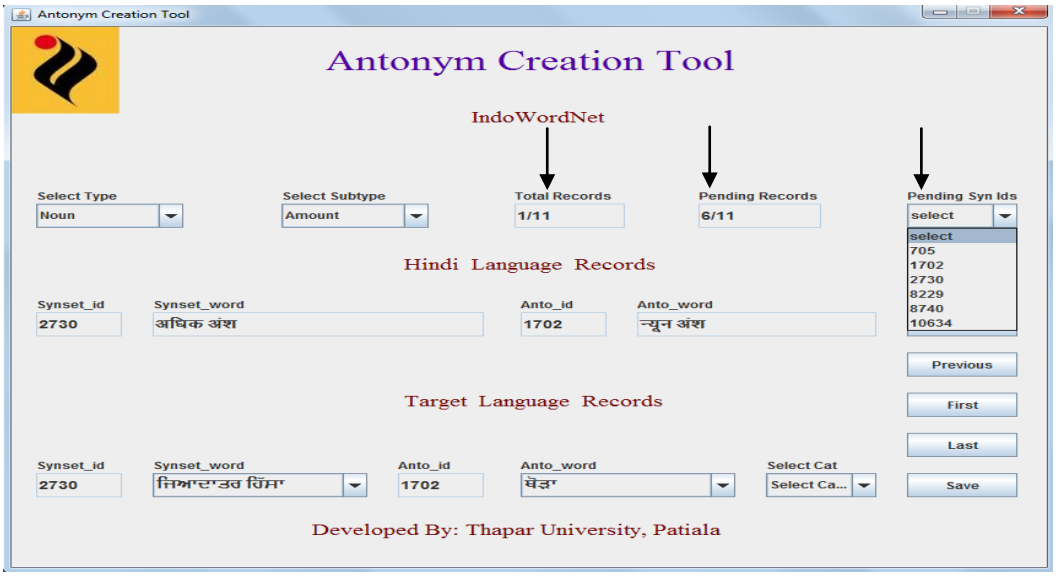

Figure 5.3: Descriptions of total records, pending records and pending synsets

The first record of the table is displayed in the textboxes under Hindi language records. It shows the ***synset Id*** of the word, word itself, ***synset Id*** of the antonym word and the antonym word. In addition to this, it also displays the synset words and antonym synset words corresponding to respective ***synset Ids*** under target language records (in this case target language is Punjabi) as shown in figure 5.4.

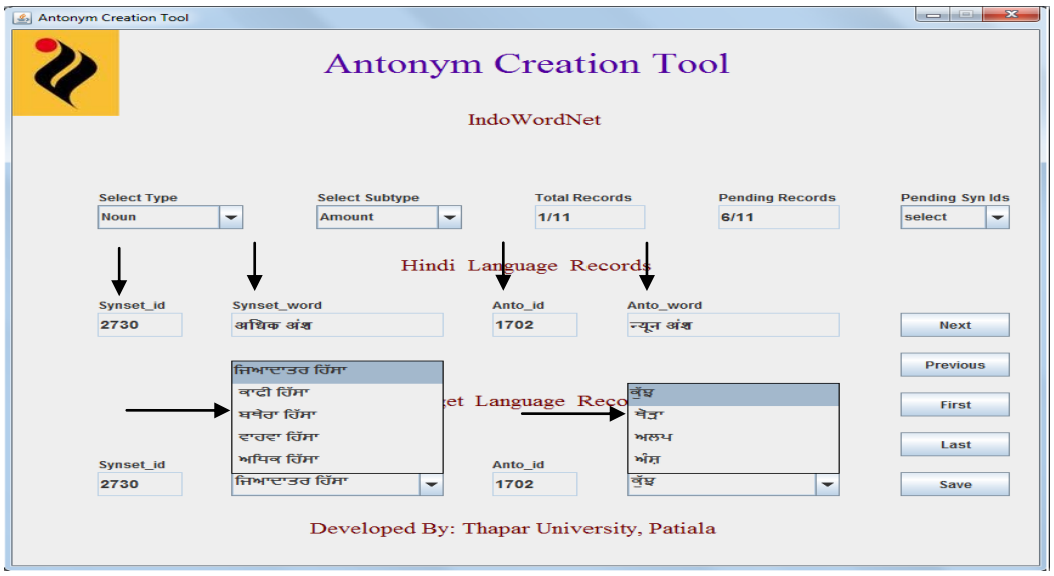

Figure 5.4: Descriptions of Hindi and target language records

Besides all these functionalities there are five buttons Next, Previous, First, Last and Save. Whenever user clicks on the next button, next record of the table is displayed in the corresponding textboxes under Hindi language records. If the corresponding ***synset Ids*** of the next record are different from the previous one, then corresponding words also change in under target language records for ***synset Id*** and antonym ***synset Id***. If target language does not have entries for the corresponding synsets, then a message is displayed no synsets found in target language as shown in figure 5.5.

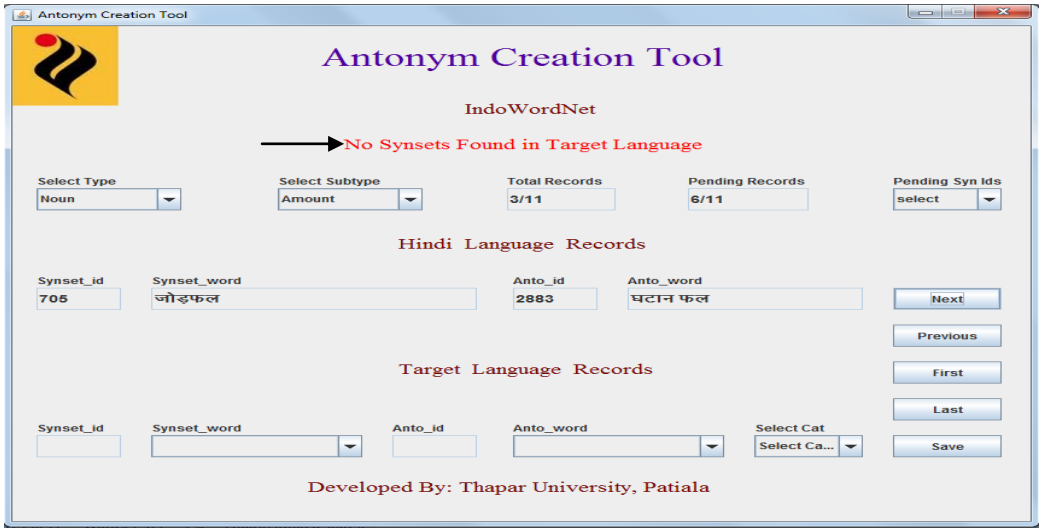

Figure 5.5: Alert message

Whenever user have to save the data for the creation antonyms in target language user can select the appropriate antonym words from the drop down lists under the target language records and select the appropriate category, then click on the save button. If record is inserted successfully it shows a message "Record Saved Successfully" as shown in figure 5.6.

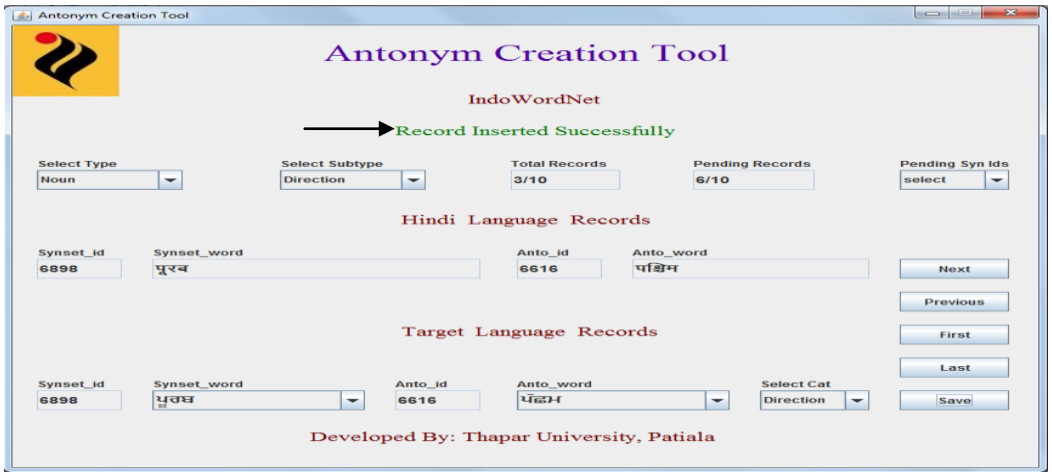

Figure 5.6: Saving records

User can also select ***synset Ids*** from the pending ***synset Ids***. It updates textboxes under Hindi language records with first word of the synset in the currently selected table and entries under target language records are updated accordingly. Textbox under total records is updated according to the number of words of the currently selected synset. It shows 1st record of currently selected synset and total number of words in the synset as shown in figure 5.7.

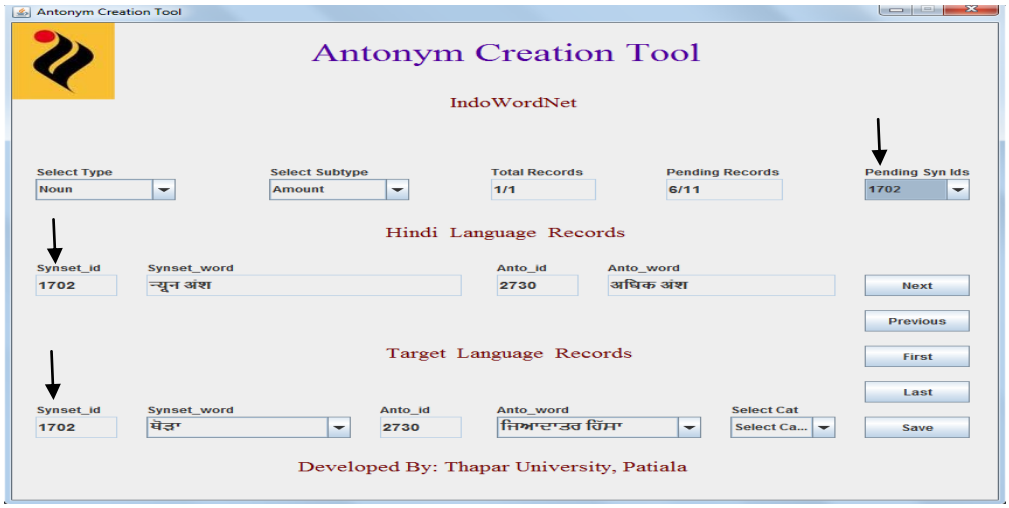

Figure 5.7: Working with pending synsets

5.2 Antonymy creation tool without referring to Hindi WordNet

In this case, we do not require any values from Hindi WordNet. Only the target language records are generated. The interface for antonymy creation tool without reference to Hindi WordNet has been designed with a provision to display the target language keyboard. For given language, a keyboard can be displayed by making changes in charfile.txt given with this tool package. User needs to enter the characters of particular language in charfile.txt and save the file. For example, in order to display the keyboard for Punjabi language enter the Punjabi characters in charfile.txt file as shown in figure 5.8 and save the file.

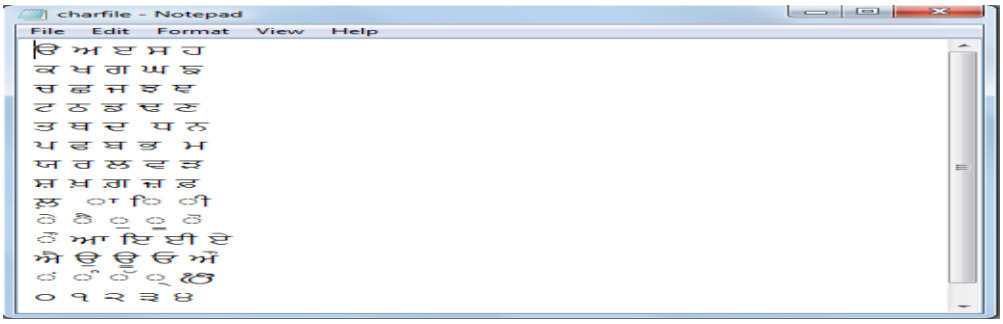

Figure 5.8: Configuring antonym creation tool

The keyboard corresponding to Punjabi language is shown in the interface as shown in figure 5.9.

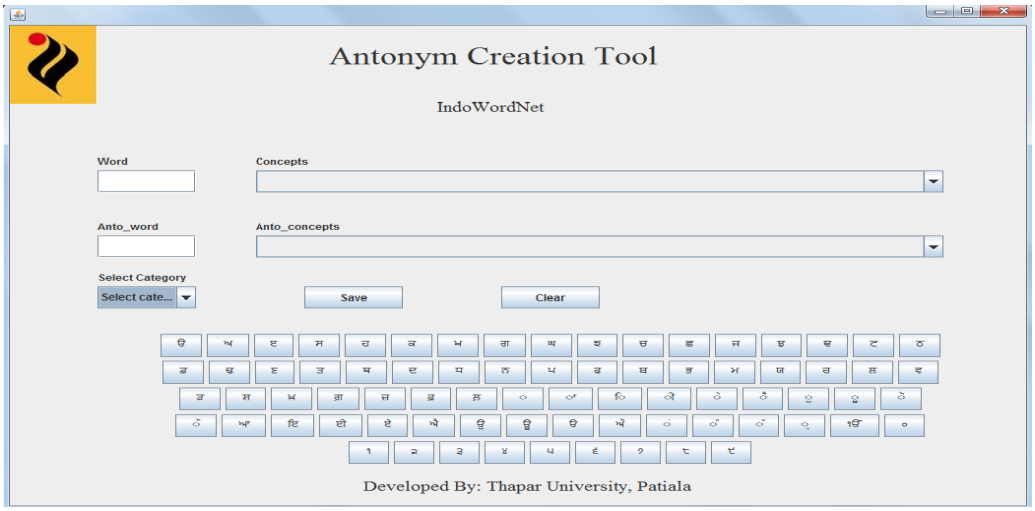

Figure 5.9: Antonym creation tool without reference to Hindi WordNet

User needs to enter a word in the textbox and press ENTER key. If the word is present in the database, the system finds the different concepts of the word in the database, otherwise a message will be displayed that word not present in the database. For example, if user enters the word ਦਿਨ *din* '*day*' and press ENTER key then it populates the drop down list with different concepts of the word ਦਿਨ *din* '*day*' as shown in figure 5.10. User can select appropriate word for the concept ਦਿਨ *din* '*day*'.

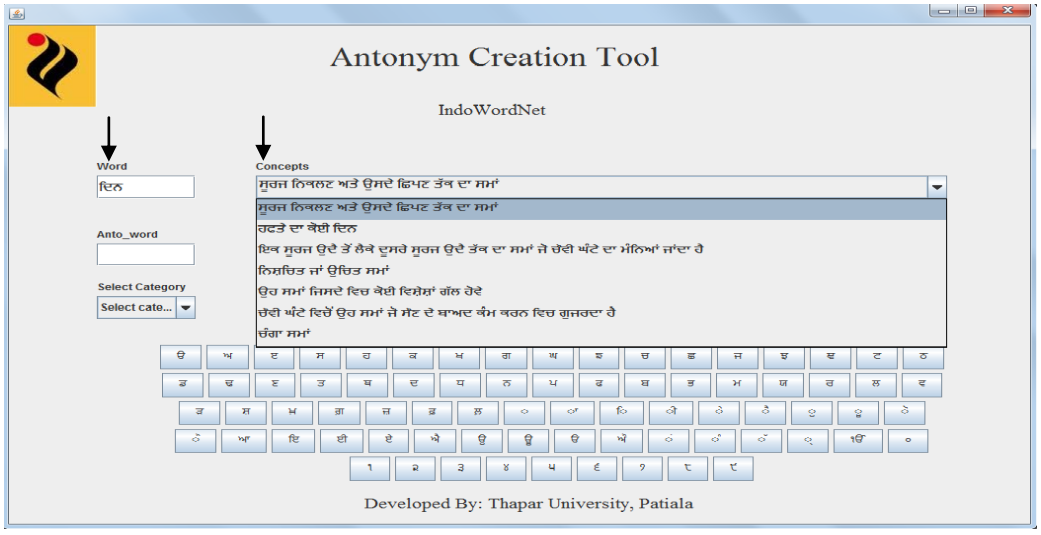

Figure 5.10: Interface showing concepts for the word ਦਿਨ

Similarly, user needs to enter the antonym word and press enter key, the system finds the different concepts for the antonym word in the database. For example, for the antonym word ਰਾਤ *rāt 'night'* the different concepts are shown in figure 5.11. User can select the appropriate concept for the word ਰਾਤ *rāt 'night'*.

Figure 5.11: Interface showing concepts for the word ਰਾਤ

After the selection of appropriate concepts for the word and antonym word, user then select the appropriate category as shown in figure 5.12 and click on save button in order to save the records.

Figure 5.12: Description of category

5.3 Compounding creation tool with reference to words in Hindi WordNet

Compounding relation relates a compound word and part word. A compound word is made when two words are joined to form a new word**.** For example, the words ਚਮੜੀ *camṛī ‘skin’* and ਰੋਗ *rōg* ‘*disease*’ joined together to form a new word ਚਮੜੀ_ਰੋਗ *camṛīrōg ‘skin disease’*. Here, ਚਮੜੀ_ਰੋਗ *camṛīrōg ‘skin disease’* is in compounding relation with the word ਰੋਗ *rōg* ‘*disease*’. An interface has been designed to create such relations from compounding relations that already exist in Hindi WordNet. The tool reduces the manual typing effort for the creation of compounding relation. The tool provides a user friendly interface for creation of compounding relations for target language. Corresponding to ***synset Ids*** of compound word and part word for Hindi language, system searches for the corresponding words in the target language. For example, corresponding to compound ***synset id*** 772 and part ***synset id*** 1423, Punjabi words are populated in the two drop down menus as shown in figure 5.13.

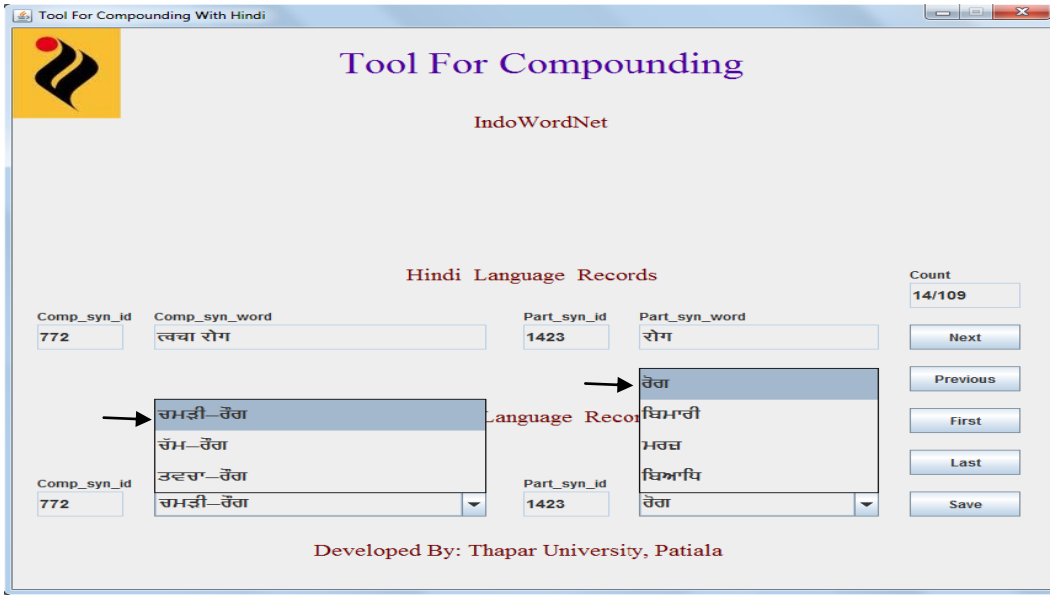

Figure 5.13: Compounding creation tool with reference to Hindi WordNet

5.4 Compounding creation tool without reference to Hindi WordNet

In this case we do not need to refer to Hindi WordNet and tool creates the compounding relations for the words which are not present in Hindi. User needs to enter a word into a text box and press ENTER key, system finds different concepts of the word. Similarly, user needs to enter part word and press ENTER key, system

searches for different concepts of the part word. User can select the appropriate concepts and click on save button in order to save the records as shown in figure 5.14.

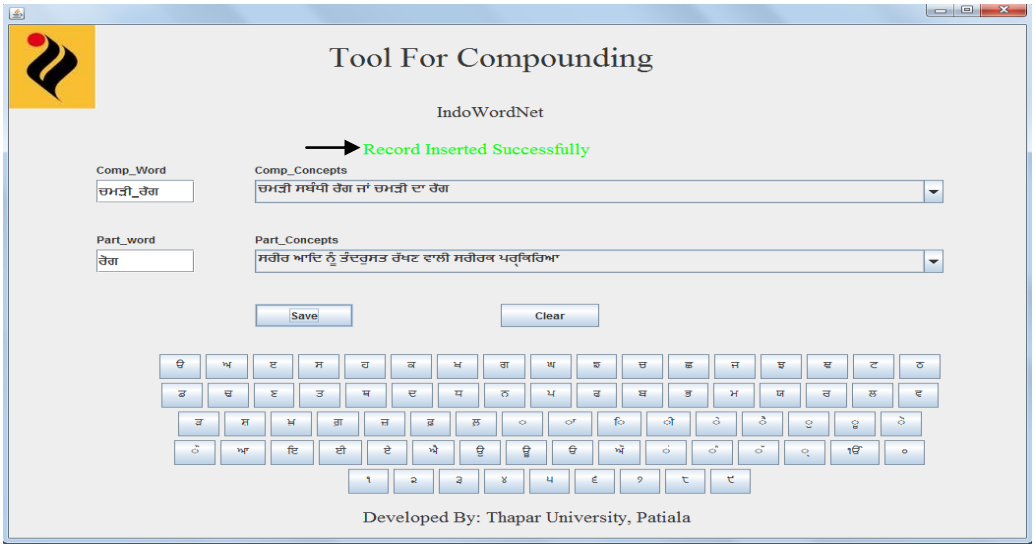

Figure 5.14: Compounding creation tool without reference to Hindi

5.5 Conjunction creation tool without reference to Hindi WordNet

Conjunction relation relates a conjunction word and its part word. Conjunction word made of two words joined together. For example, the word ਪੂਜਾ_ਕਰਨਾ *pūjākarnā 'pray'* is made of two words ਪੂਜਾ *pūjā 'pray'* and ਕਰਨਾ *karnā 'to do'*. Here, ਪੂਜਾ_ਕਰਨਾ *pūjākarnā 'pray'* is in conjunction relation with word ਪੂਜਾ *pūjā 'pray'*. The basis of this tool is the conjunction relation for Hindi language. Corresponding to ***synset ids*** of the conjunction word and part word in Hindi language the system searches for the words in the target language. Now, user can select the appropriate conjunction word and part word from the two drop downs for the creation of conjunction relation for the target language. For example, corresponding to conjunction ***synset id*** 3047 and part ***synset id*** 3046, Punjabi words are populated in the drop down menus as shown in figure 5.15.

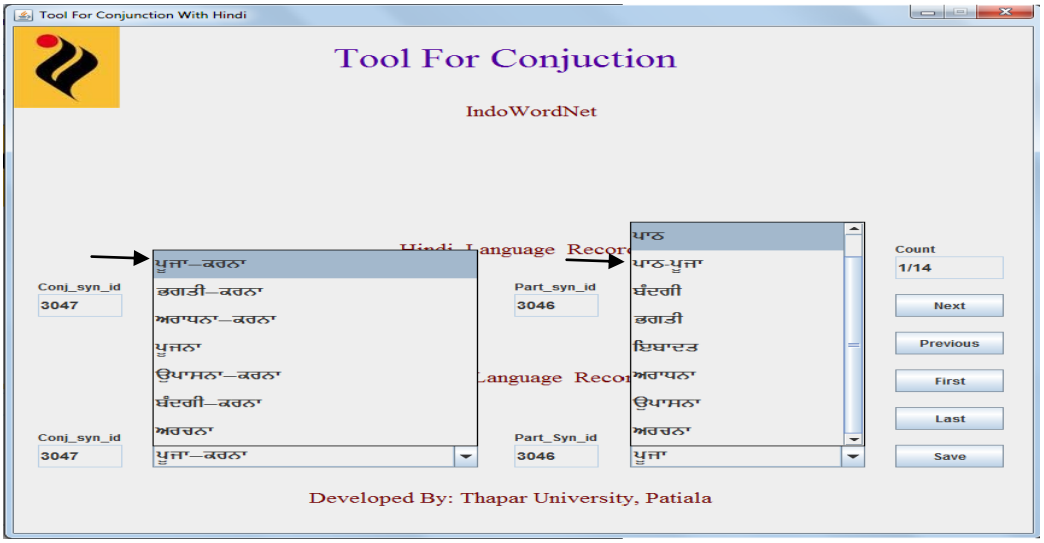

Figure 5.15: Conjunction creation tool with reference to Hindi WordNet

5.6 Conjunction creation tool without reference to Hindi WordNet

Conjunction relation may also exist between the words which are not present in Hindi WordNet, but present in target language. An interface has been designed to create conjunction relations for the words which are not present in Hindi language. User needs to enter a word into a text box and press ENTER key, a drop down menu is populated with different concepts of the word. Similarly, user needs to enter part word and press ENTER key, another drop down is populated with the different concepts of the part word. User needs to select the appropriate concepts and click on save button in order to save the records as shown in figure 5.16.

Figure 5.16: Conjunction creation tool without reference to Hindi WordNet

5.7 Gradation creation tool with reference to the words in Hindi WordNet

Gradation is lexical relation that exists between three word forms. It represents the intermediate concept between two opposite concepts. For example, the words ਸਵੇਰ *savēr 'morning'*, ਦੁਪਿਹਰ *dupihar 'noon'* and ਸ਼ਾਮ *shām 'evening'* show the concept of gradation. The words ਸਵੇਰ *savēr 'morning'* and ਸ਼ਾਮ *shām 'evening'* are opposites of each other and ਦੁਪਿਹਰ *dupihar 'noon'* is the intermediate concept between these antonyms. The basis of this tool is the gradation relations for Hindi language. Corresponding to the synset ids obtained for Hindi language for first, mid and last word, the system finds the words for the target language. For example, corresponding to first synset id 4859, mid synset id 8249 and last synset id 6810 system populates the drop downs with Punjabi words as shown in figure 5.17. User can select the best match from the three drop downs for the creation of gradation relation.

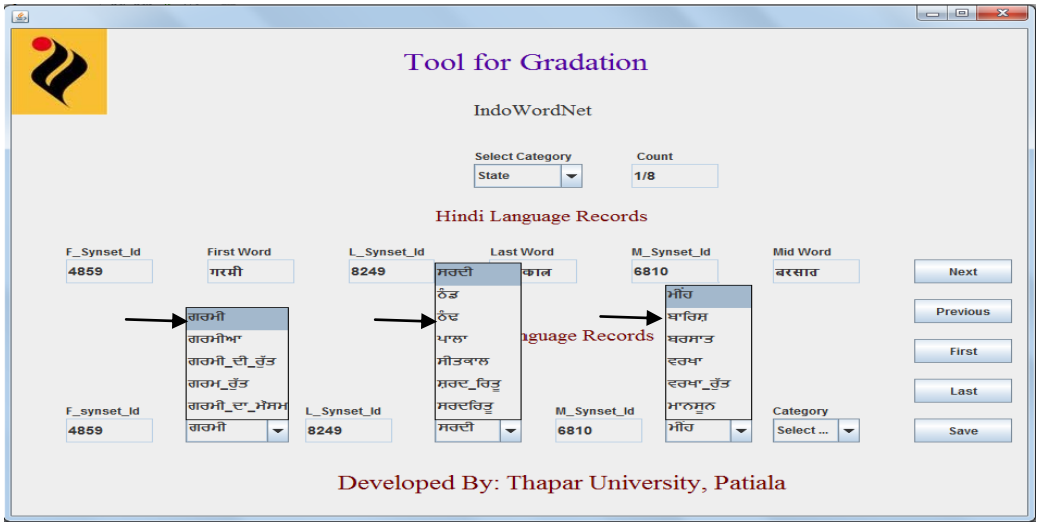

Figure 5.17: Gradation creation tool with reference to Hindi Wordnet

5.8 Gradation creation tool without reference to Hindi WordNet

The tool creates the gradation relations for words which are not present in Hindi. User needs to enter a word into a text box and press ENTER key, the system finds the different concepts of the word. Similarly, system searches for the related concepts for the mid word and last word. User needs to select the appropriate concepts and category then click on save button in order to save the records. For example, user enters the words ਬਚਪਨ *bacpan 'childhood'* ਜਵਾਨੀ *javānī 'young'* and ਬੁਢਾਪਾ *buḍhāpā*

'*old age'* system populates the corresponding drop downs with related concepts. User then select the appropriate concept and category as shown in figure 5.18 and then click on save button in order to save records.

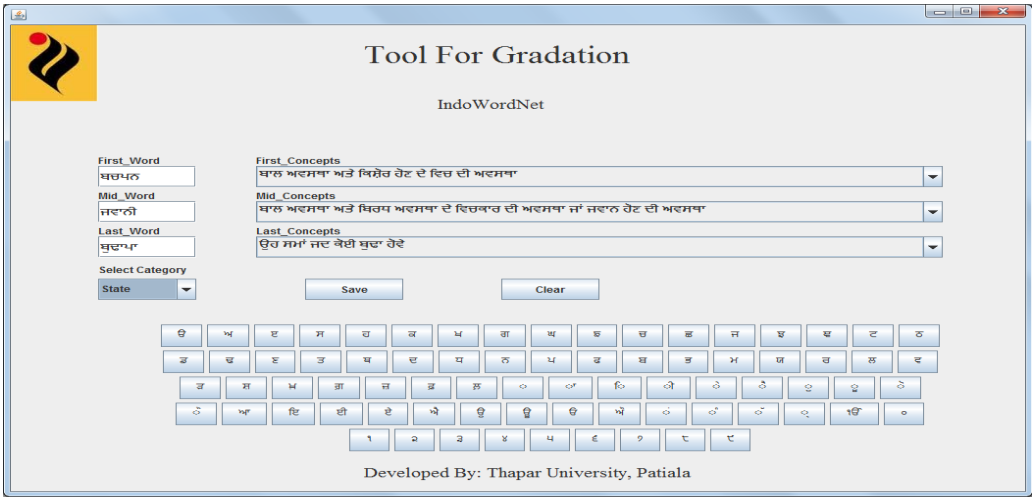

Figure 5.18: Gradation creation tool without reference to Hindi WordNet

5.9 Web interface for Punjabi WordNet

A web interface for Punjabi WordNet has been developed with JSP as front end and MySql as back end. Figure 5.19 depicts web interface for Punjabi WordNet.

Figure 5.19: Web interface for Punjabi WordNet

Punjabi keypad is used to enter the word which is to be searched. Then user clicks on search button to find various ***synset_ids*** of the word. For example, the various ***synset_ids*** for the word ਘਰ *ghar 'house'* are shown in figure 5.20.

Figure 5.20: Different senses for the word ਘਰ as noun

User can click on the ***synset_id*** in order to view concept, example, category information and synonym words as shown in figure 5.21. Moreover, it also displays the various semantic relation in which particular ***synset_id*** is involved.

Figure 5.21: Synset information for particular sense of word ਘਰ

The figure 5.21 shows that the ***synset_id*** 1901 is involved in hypernymy, hyponymy, holonymy and meronymy. In order to view the details regarding hypernymy relation

for a ***synset_id*** 1901 click on the link hypernymy. The various hypernymy relations for ***synset_id*** 1901 are shown in figure 5.22.

Figure 5.22: Hypernymy relations for particular sense of the word ਘਰ

Similarly, in order to view the hyponymy, meronymy and holonymy relations for the word ਘਰ *ghar 'house'* click on the corresponding links. The various hyponymy for the particular sense of the word ਘਰ *ghar 'house'* are shown in figure 5.23.

Figure 5.23: Hyponymy relations for particular sense of the word ਘਰ

In this chapter, working of various lexical creation tools and web application for Punjabi WordNet has been described. The conclusion and future scope of this lexical resource is presented in next chapter.

Chapter 6
Conclusion and Future Scope

6.1 Conclusion

Punjabi WordNet can be used as a lexical resource for natural language processing tasks for the Punjabi language. We believe that the work presented here is a step in the direction towards the achievement of natural language processing tasks for the Punjabi language. To eliminate the barrier of mankind's communication, the ultimate goal of constructing WordNet is to link all languages in the world together. Punjabi WordNet lexical database can be used as a lexical information source for many applications such as word sense disambiguation, information retrieval, text classification and machine translation. WordNet has been developed for many languages starting with English language, then EuroWordNet and WordNet for Indian languages is currently in progress. Punjabi WordNet is being developed from Hindi WordNet using expansion approach. IL-MultiDic development tool has been used for creation of synsets for Punjabi language with reference to Hindi WordNet. Database has been designed to store the synsets and to provide storage for lexical and semantic relations. DB Import tool has been effectively used to port the data from text files to designed database. Using expansion approach semantic relations are borrowed from the source language as they are same for all the languages. Lexical relations are language specific, so they cannot be borrowed from the source language. A lexical creation tool has been developed to create the lexical relations like antonymy, compounding, conjunction and gradation *etc*. for Punjabi language. A web application Web application for Punjabi WordNet has been developed to provide complete information about a word in the form of gloss, example sentence, synonym words and category. Web application also has a provision to show semantic relations like hypernymy, hyponymy, meronymy, holonymy, causation, entailment and troponymy *etc.*

6.2 Future scope

There are many possible extensions of this work that can be undertaken in further research. Some of them are given below.

- At present, only semantic relations like hypernymy, hyponymy, meronymy, holonymy, troponymy, entailment and causation *etc.* have been included in the Punjabi WordNet.

- Database for this lexical resource can be tuned to add more relations.
- The performance can surely be improved if morphology is handled exhaustively. The system currently does not detect the underlying similarity in presence of morphological variations.
- This lexical resource for Punjabi can be used for testing of various algorithms related to word sense disambiguation problem.
- WordNet can be easily extended to similar languages of the same family.
- Applications of this lexical resource can be extended to various natural language processing tasks.

References

[1] Ahmed T. and Hautli A., "A first approach towards urdu WordNet," *J. of Language and Literature Review*, vol. 6, no. 1, 2011 pp: 1-14.

[2] Banerjee S. and Pedersen T., "An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet," in *Proc. Third International Conference on Intelligent Text Processing and Computational Linguistics*, Mexico City, 2002, pp: 1-10.

[3] Banerjee S. and Pedersen T., "Extended Gloss Overlaps as a Measure of Semantic Relatedness," in *Proc. Eighteenth International Joint Conference on Artificial Intelligence*, Mexico, August, 2003, pp: 1-6.

[4] Bhattacharya P., "IndoWordNet," in *Proc. Lexical Resources Engineering Conference,* Malta, 2010, pp: 1-8.

[5] Bhensdadia C.K. *et. al.*, "Introduction to Gujarati WordNet," in *Proc. Third National Workshop on IndoWordNet*, IIT Kharagpur, 2010, pp: 1-5.

[6] Brent, M. R., "From grammar to lexicon: Unsupervised learning of lexical Syntax," *J. Computational Linguistics*, vol.19, no.2, 1993, pp: 1-20.

[7] Database design by Goa University, Goa, Tech. Rep., 2011, pp: 1-13.

[8] English WordNet by Princeton University [online], Available: http://wordnetweb.princeton.edu/perl/webwn [Accessed: June 06, 2012].

[9] El-Kahlout D. and Oflazer K., "Use of Wordnet for Retrieving Words from Their Meanings," in *Proc. of the Second Global WordNet Conference*, Czech Republic, 2004, pp. 118-123.

[10] Farque F. and Khan M., "BWN - A Software Platform for Developing Bengali WordNet," in *Proc. International joint conferences on computer, information, system sciences and engineering* (CISSE), 2008, PP: 337-342.

[11] Gabrilovich E. and Markovitch S.**,** "Text Categorization with Many Redundant Features: Using Aggressive Feature Selection to Make SVMs Competitive with C4.5," in *Proc. 21st International Conference on Machine Learning*, Canada, July 2004, pp: 321-328.

[12] Hindi WordNet from CFILT, IIT Bombay, India [Online]. Available: http://www.cfilt.iitb.ac.in/WordNet/webhwn [Accessed: June 06, 2012].

[13] Hindi WordNet from CFILT, IIT Bombay, Tech. Rep., 2007, pp: 1-8.

[14] Kaur R. *et. al*, "Punjabi WordNet Relations and Categorization of Synsets*,"* in *Proc. Third National Workshop on IndoWordNet*, IIT Kharagpur, 2010, pp: 1-6.

[15] Kulkarni M. *et. al*, "Introducing Sanskrit WordNet," in *Proc. Global WordNet Conference*, IIT Mumbai, 2010, pp: 1-8.

[16] Miller G. A., *et. al*, "Introduction to WordNet: An On-line Lexical Database," *International J. of Lexicography*, 1990, pp: 235-244.

[17] Morato J. *et. al*, "WordNet Applications," in *Proc. of the Second Global WordNet Conference*, Czech Republic, 2004, pp. 270-278.

[18] Ramakrishnan G. *et. al*, "Soft word sense disambiguation," in *Proc. Global WordNet Conference*, Czech Republic, 2004, pp: 1-9.

[19] Rana S., "Punjabi WordNet –A tool for Natural Language Processing," ME Thesis, Computer Science and Engineering Department, Thapar University, Patiala, 2007.

[20] Rattan R, "Creation of Punjabi WordNet and Punjabi Hindi Bilingual Dictionary," ME Thesis, Computer Science and Engineering Department, Thapar University, Patiala, 2007.

[21] Rattan R. *et. al*, "Design and development of Punjabi WordNet and Punjabi Hindi bilingual dictionary," in *Proc. Second National level Indradhanush WordNet workshop,* Goa University, Goa, August, 2010, pp: 1-6.

[22] Sarma S. K. *et. al*, "Foundation and Structure Of Developing An Assamese WordNet," in *Proc. Global WordNet Conference*, 2010, pp: 1-5.

[23] Sarma S. K. *et. al*, "A WordNet For Bodo Language: Structure And Development," in *Proc. Global WordNet Conference* 2010, pp: 1-4.

[24] Tufis D. *et. al*, "Balkanet: Aims, methods, results and perspectives. A general overview," *Romanian J. Sci. Tech. Inform.* vol.7, no.1, 2004, pp: 9-43.

[25] Verma N. and Bhattacharyya p., "Automatic Lexicon Generation through WordNet," in *Proc. of the Second Global WordNet Conference*, Czech Republic, 2004, pp: 226-233.

[26] Vossen P., "EuroWordNet: A multilingual Database with Lexical Semantic Networks," *J. of computational Linguistics*, vol.25, no.4, 1998, pp: 628-630.

[27] Walawalikar S. *et. al*, "Experiences in Building the Konkani WordNet Using the Expansion Approach," in *Proc. Global WordNet Conference*, 2010, pp: 1-7.

[28] Zaiane O. R. *et. al*, "Word taxonomy for online visual asset management and mining, application of Natural Language to Information Systems," in *Proc. 4th International Conference NLDB*, Vienna, 1999, pp: 271-275.

Research Papers Communicated

- Ashish Narang and Parteek Kumar, "Lexico semantic relations for Punjabi WordNet," communicated to *Telecommunications Journal*.

- Ashish Narang and Parteek Kumar, "Creation of Lexical relations for Punjabi WordNet," communicated to *International Journal of Computer Applications*.