# **Chapter 1**

# **Text To Speech**

# **Introduction**

Speech is the primary mode of communication between humans. Speech Synthesis is an artificial production of the human speech. Speech Synthesis is the process, which allows transformation of the string of phonetic and prosodic symbol into a synthetic speech signal. TTS technology is the branch of artificial intelligence. TTS system is a process through which input text is analyzed, processed, and understood and the text is rendered to digital audio and then spoken.

Text to speech (TTS) synthesis is the automated transformation of a text into speech that sounds as close as possible, as a native speaker or the language reading the text. Most Text To Speech Systems can be categorized by the method that they use to translate phonemes into audible sound. Some of them are Prerecorded, Formant, Concatenated, etc. The following table shows the comparison

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Prerecorded** | **Concatenated** | **Formant** |
| **Resource**  **requirement** | Very large storage, Small memory | Large storage, Very large memory | Low storage, relatively small memory |
| **Vocabulary** | Limited | Unlimited | Unlimited |
| **Voice quality** | Natural, Most pleasant | Natural | Robotic, Sometimes not appreciated by the user |
| **Multiple**  **featured voices** | Need high storage in that case | Need high storage in that case | Can produce multiple featured voices without any major changes |
| **Intelligibility** | High | Highly Intelligible | High |

*Table 1: Comparison between different categories of TTS*

All the TTS system takes the text in digital format, such as ASCII for English, Unicode for Nepali. The quality of result is a function of the quality of the text, as well as of the quality of the generation process itself. The first requirement of TTS system is intelligibility and the second one is naturalness.[1, 2]

Nepali text to speech (TTS) is an application for converting the text documents into the audio files that is played with a media player with effective speech for listening. This software can read Nepali text from any documents, files, and websites or self inputs for the people and the file will be generated in WAV audio format. A TTS system has text fed as input to generate synthetic speech waveform as the output for the corresponding text through an audio device to create the speech sounds.

*Fig 1.1: Schematic Representation of text to speech (TTS) system*



Speech

**Input:** Text in any language

**Text-To-Speech System**

**Output:** Audio

The TTS system is becoming more interactive and helpful to the users, especially physically and visually impaired and illiterate masses, i.e. not everyone can read text when displayed on the screen or when printed this may be because the person is partially sighted, or because they are not literate and these people can be helped by generating speech rather than printing or displaying it, using TTS system to produce the speech for the given text[3, 4]. So that the TTS synthesis has a great demand for Nepali language.

Speech application are primarily to simplify and automate tasks for human to make it easier assisting them in resolving certain problem domain. TTS synthesis has a wide range of application in everyday life because no longer people want to sit and read data from the monitor. Since there is painstaking efforts to be taken, this involves strain to theirs eyes.[18]

**NLP**

Linguistic Representation Phasing Intonation Duration

**DSP**

Waveform generation

**Text To Speech Synthesizer**

Phonemes

Prosody

Text

Speech

*Fig 1.2: A simple functional diagram of TTS system*

A closure examination of TTS system problem of converting text into voice output can be subdivided into two phases [5]. The first phase is Natural Language Processing (NLP)[17] or high level which involves proper analysis of text with linguistic rules in the language. This analyses the input text and converts the orthographic representation of the text into its linguistic representation. The linguistic representation gives information about grammatical categories of words, their tonal properties and most importantly pronunciation of the words.

The second phase is Digital Signal Processing (DSP) or low level which involves proper speech synthesis, i.e. generation of speech waveform from the internal linguistic representation.

Speech can be synthesized by creating an environment of oral track simulation (articulatory synthesis) or by creating an acoustic model (formant synthesis) or by manipulating the prerecorded speech units (concatenative synthesis).

## **Problem statements**

Until now, there have been development of text to speech systems in language like English, German, Hindi and others. There has only been one or two attempt to build Nepali TTS.

The problem domain could be of disable people who are vocally impaired, those people heard the voice that is generated by the Text-To-Speech Synthesizer. The people who are visually impaired cannot read the normal language except Braille. There are no such application yet so far to assist them. Currently, no any TTS system for Nepali language have been developed till now except Bhashasanchar. So, it is hard to find the research work, documentation and the source to get help from for building the TTS.

## **Aims and Objective**

The aim of the project is to develop the Nepali speech synthesizer that allows generating the speech output from the input Nepali text.

The main objectives of Text To Speech system are:

* To design and develop a TTS system and to implement the system that can generate the audio output from the input Nepali Text.
* To resolve inability of people in Nepal who are illiterate to understand what the bunch of characters in front of them mean and to provide ease of use for disable persons with visual impairments.

## **Application**

The TTS system has many applications. The application of synthesized speech ranges from small “talking clock” to advance talking person (a multimedia application). In the domain where limited vocabulary is needed, such talking calculator or clock, one can build a simple speech synthesizer specific to that domain only. Where in the applications where the vocabulary is not limited, such as a web page reader or email-over-telephone reader, one has to build a TTS system with unlimited vocabulary in consideration.

* **Application for visually impaired**

The most useful application of speech synthesis that comes into one’s mind is to help visually impaired people read and communicate. There are so much information in the internet and in printed form. But people with visual problem can’t read those texts. With a text-to-speech system we could build application which can read out texts from computer. Actually there have been efforts to build software which are able to read texts typed in the computer.

* **Application for vocally handicapped**

People who are vocally handicapped find themselves difficulty in communicating with other people who do not understand their way of communicating i.e. using sign language. This synthesizer gives them the opportunity to communicate easily with other people who do not understand the sign language. Using this tool they can convey any message they want to, to the community. This can improve quality of the communication between the normal people and the handicapped people.

This kind of users may also be frustrated by an inability to convey their emotions such as happiness, sadness, urgency, or friendliness by voice. For this we can make use of different tags for different kinds of emotions.

For example for showing urgency through voice we can make use of urgent tag in front of urgent sentences. This tag helps the synthesizer read the text quicker which makes other feel that the user's message is urgent one.

Similarly, for sadness we can make use of sad tag for which the message given by the user will be read in slow voice by the speech synthesizer. For conveying urgent message the following syntax for giving input text will work.

* **Telecommunication application**

The latest applications of Nepali Speech Synthesizer are in multimedia application. Synthesized speech has been used for many years in telephone enquiry systems. Today, the quality of synthesized speech has increased so much that normal customers are adopting for daily use. Electronic mails are the most popular way of communicating used by most of the people of the world in last few years.

However, sometimes it is not possible to read those e-mail messages in our inbox when we are busy. With Speech Synthesizers it is possible to read the email messages via normal telephone also.

This also helps for those people who can understand but can not read and write the Nepali language. Synthesized Speech may also be used to speak out short text messages (SMS) in mobile phones.

This system is also helpful in reading out Nepali messages from your email inbox. This system is used as an interface to record different voice messages used for mobile phones, landlines etc.

* **Language Education**

This project is targeted on Nepali text. Around 60% of Nepalese are illiterate. TTS can provide a very handy tool to educate people about the language.

# **Chapter 2**

# **Literature Review**

## **2.1 History of Speech Synthesis**

Researchers have been studying for centuries for artificial production of speech[6]. The effort has transited from mechanical modeling of human speech production system to electrical speech synthesizers and now to different modern synthesis techniques of concatenating recorded speech with text analysis to obtain more natural sounding voice outputs.

The earliest efforts of producing artificial sound were with different mechanical devices which model the human speech production system. The earliest mechanical models had different music instrument like devices capable of producing only five long vowels each (a, e, i, o, and u). To produce voice from these devices air had to be blown with air. Hence the production of voice was limited to very basic vowels and the process of speech production was not automatic. These were the acoustic resonators modelling the human vocal tract. After a few generations other parts of the machine were improved; like pressure chamber for lungs, vibrating reed to act like vocal cords and leather tube for the vocal tract. With appropriate manipulation of the shape of the leather tube these machines were able to produce vowel sounds and consonants were simulated by separate constricted passages and controlled by the fingers. To simulate more sounds other components were added to the machine, e.g. movable lips, and tongue. Such machines were a good model of the human speech system but were limited to produce phones and limited set of words only, but not long sentences.

First full electrical synthesis devices had a buzzer as excitation and resonant circuits to model the acoustic resonances of the vocal tract. These were able to generate single vowel sounds and no consonants. The first true speech synthesizer was introduced by Homer Dudley in 1939, called VODER (Voice Coder). The VODER consisted of wrist bar for selecting a voicing or noise source and foot pedal to control the fundamental frequency. Source signal was passed through ten bandpass filters whose output levels were controlled by fingers. Only skilled operator of VODER could produce a sentence of speech from the device. The demonstration of VODER showed that artificial production of speech was possible and increased more interest towards speech synthesis.

Further study on speech signal and its decomposition invented new technique of speech synthesis called *formant synthesis* with proper prediction of parameters representing the signal. Formant synthesis does not produce natural sounding speech when operated in fully automated mode to predict the signal parameters. With the advent of digital representation of digital sounds, availability of cheap and powerful computer hardware, and different digital signal processing techniques speech generation method shifted from fully synthetic to concatenation of natural recorded speech. Speech generated from concatenation method resulted more closely to natural voice than the fully synthesized voice. In addition to this, since memory cost has been dramatically decreased and the processing speed has been exponentially increased the developers are nowadays interested in different concatenative approaches.

## **2.2 TTS for Nepali**

The first Nepali Text-to-Speech synthesizer that was built in 2000 by Sameer K. Maskey while he was doing his under graduation course in Carnegie Mellon University (CMU). The system was based on the Festvox tool built by Alan Black at CMU, US. A presentation of the software was also delivered at RONAST (Royal Nepal Academy of Science and Technology). Unfortunately, this first Nepali Text-to speech systems appears to have been lost.

The development of Nepali TTS at MPP began in 2005 with some preliminary reading about Festival and Festvox, and an introduction to speech in Sweden in September 2005. It started in earnest in Jan 2006 with a team of two software engineers (Srishtee Gurung and Ishwor Thapa) and part-time linguist suport (from Madhav Pokharel and Bhim Regmi). Gurung and Thapa visit the Speech Lab at the International Institute of Information Technology (IIIT), Hyderabad in February 2006, working under the guidance of Kishore Prahallad. They produced the first prototype of Nepali voice, a male voice, during the visit. The experience and knowledge gained in one month training gave them a solid background to produce synthetic speech for Nepali language. After this visit support came mostly from the mailing list of the Festival Speech Synthesis System, particularly for building letter-to-sound rules. By the end of 2006 they had produced Nepali voices for both male and female implemented as a Web based application. This application can be used to read any Nepali text given as an input. In addition to it, in order to produce support for visually impaired and non-literate people the Nepali Voice has been integrated with the Ubuntu version of Linux to produce a Screen Reader – producing a screen reader for Neplainux will come in 2008, but there are major difficulties in producing a screen reader for Windows.

Nepali Text-to-Speech (NTTS) is a synthetic voice generation system. It can produce Nepali voice from any Nepali text written in Unicode (UTF-8) format. We believe that our TTS though primarily developed for general audience, will be of great assistance to the disabled and non-literate communities of the society who have been put aside in current technology in Nepal. [1]

### **2.2.1 Bhashasanchar NeLRaLEC Project**

Bhashasanchar NeLRaLEC Project [6] or Nepali TTS is being developed using the framework of Festival Speech Synthesis System developed by University of Edinburgh. This is free software which supports multi-lingual speech synthesis and has an open architecture for research in this field. The different phases for the development of this project are:

**Phase 1: Analysis of Nepali Language**

* Defining Nepali Phone set
* Defining stress, syllabification and schwa (insertion- deletion) rules
* Defining a letter to sound rules
* Defining a lexicon for Nepali TTS
* Part of Speech (POS) tagging for homograph disambiguation
* Text Normalization

**Phase 2: Building Synthetic Voices**

* Optimal Text Collection
* Recording of Speech (one male and one female)
* Speech Labeling
* Extracting the speech parameters
* Building utterances and synthesizing
* Playing the voice
* Tuning the voice

The project’s main activities are:

* Gather text and speech corpora representing examples of written and spoken texts of literary and other genres available in Nepali language, to form the foundation for all subsequent activities.
* Produce industrial- strength localized basic software for Nepali.
* Produce a corpus-based dictionary of contemporary Nepali and develop basic language processing capabilities for Nepali.
* Produce speech technologies for automatically reading Nepali text.
* Disseminate technologies to school and universities and local communities and evaluate their use there.

## **2.3 Text to Speech Systems for Indian Languages**

In India, to help the visually impaired, vocally disabled and day to day increasing applications of speech synthesis has necessitated the development of more and more innovative text-to-speech (TTS) system. Some of the already developed TTS are described below. In this paper four Indian languages text-to-speech systems, namely Dhvani, Shruti, HP Lab system based on Festival framework.

**2.3.1 Dhvani- Indian Language Text-to-Speech System**

Dhvani [7] system has won FOSS India award in 2008. The main characteristics of this system are as follows.

* Dhvani is a Text To Speech System specially designed for Indian languages.
* This system has been developed by Simputer trust headed by Dr. Ramesh Hariharan at Indian Institute of Science Bangalore in year 2000.
* It uses diphone concatenation algorithm.
* Currently this system has Hindi, Malayalam, Kannada, Bengali, Oriya, Punjabi, Gujarati, Telugu and Marathi modules.
* All sound files stored in the database are ‘gsm’ compressed files.
* It has different modules for every language.
* It is based on the observation that a direct grapheme to phoneme mapping exits for all Indian languages in general.
* It is an attempt in India to cover all Indian languages under a single framework.
* In this system each language requires a Unicode parser.

**2.3.2 Architecture of Dhvani system**

The architecture of **Dhvani** system is described as below. It includes the following components.

1. **Text parser**: Each language requires a Unicode parser. It simply parses the input text.
2. **Text toDhvani phonetic script:**This makes **Dhvani**, language independent. In this phase any Unicode text is converted to a common script. This script is further act as the input to the speech synthesizer.
3. **Grapheme to phoneme conversion:**This converts the grapheme to phoneme. The phonetic description is syllable based. In this eight kinds of sounds are allowed.
4. **Sound database:**All sound files stored in the database are *‘gsm’* compressed files.
5. **Speech synthesizer:**Speech synthesizer takes phonetic script and with the help of CV(consonant-vowel) pair algorithm and sound concatenation component concatenates the sound files to produce speech.

# **Chapter 3**

# **Methodology**

## **3.1 Techniques of Speech Synthesis**

The two characteristics used to describe the quality of a speech synthesis[8] system are naturalness and intelligibility. The naturalness of a speech synthesizer refers to how much the output sounds like the speech of a real person. The intelligibility of a speech synthesizer refers to how easily the output can be understood. The ideal speech synthesizer is both natural and intelligible, and each of the different synthesis technologies try to maximize both of these characteristics. Some of the technologies are better at naturalness or intelligibility and the goals of a synthesis system will often determine what approach is used. There are few main technologies used for the generating synthetic speech waveforms: concatenativesynthesis, formant synthesis, articulary synthesis and some are described below*.*

## **3.1.1 Formant synthesis**

Formant synthesis[9] does not use human speech samples at runtime. Instead, the synthesized speech output is created using additive synthesis and an acoustic model. Parameters such as fundamental frequency, voicing and noise levels are varied over time to create a waveform of artificial speech. This method is sometimes called rule based synthesis; however many Concatenative systems also have rule based components. Formant synthesizers are usually smaller programs than Concatenative systems because they do not have a database of speech samples. This is also called the terminal analog model.

Many systems based on formant synthesis technology generate artificial, robotic-sounding speech, and the output would never be mistaken for the speech of a real human. However, maximum naturalness is not always the goal of a speech synthesis system, and formant synthesis systems have some advantages over concatenative systems.

Formant synthesized speech can be very reliably intelligible, even at very high speeds, avoiding the acoustic glitches that can often plague concatenative systems. First high speed synthesized speech is often used by the visually impaired for quickly navigating computers using a screen reader. Second, formant synthesizers are often smaller programs than concatenative systems because they do not have a database of speech samples. They can thus be used in embedded computing situations where memory space and processor power are often scarce. Last, because formant-based systems have total control over all aspects of the output speech, a wide variety of prosody or intonation can be output, conveying not just questions and statements, but a variety of emotions and tones of voice. Probably the most widely used synthesis method during last decades has been formant synthesis. [16]

## **3.1.2 Concatenative Synthesis**

Concatenative synthesis [10] is based on the concatenation of segments of the recorded speech, and possible modification of prosody such as modulation and duration. Generally, Concatenative synthesis produces the most natural-sounding synthesized speech. Word level concatenation is not a viable since it requires recording of large level of units. Concatenative synthesis is probably the easiest method to produce intelligible and natural sounding synthetic speech. In Concatenative synthesis we connect prerecorded natural utterances to produce new natural sounds. However concatenative synthesizers are limited to one speaker and one voice and usually require more memory than other methods. The most important aspect in concatenative synthesis is to find correct unit length. We had to make selection between longer units and shorter units. Longer units possesses high naturalness, contains less concatenation points and more control over the variation in phonemes (co-articulation) can be achieved. But the amount of required units and memory goes on increasing. With Shorter Units less memory is needed, but the sample collecting and labeling procedures become more difficult and complex. In present systems units used are usually words, syllables, demi-syllables, phonemes, diphones, and sometimes even triphones.

There are two approaches of the Concatenative synthesis: Diphone-based synthesis and unit selection synthesis.

**Unit selection synthesis**uses large speech databases. During database creation, each recorded utterance is segmented into some or all of the following: individual phones, syllables, morphemes, words, phrases, and sentences. Typically, the division into segments is done using a specially modified speech recognizer set to a "forced alignment" mode with some hand correction afterward, using visual representations such as the waveform and spectrogram. An index of the units in the speech database is then created based on the segmentation and acoustic parameters like the fundamental frequency (pitch), duration, position in the syllable, and neighboring phones. At runtime, the desired target utterance is created by determining the best chain of candidate units from the database (unit selection). This process is typically achieved using a specially-weighted decision tree. Unit selection gives the greatest naturalness due to the fact that it does not apply a large amount of digital signal processing to the recorded speech, which often makes recorded speech sound less natural, although some systems may use a small amount of signal processing at the point of concatenation to smooth the waveform. In fact, output from the best unit selection systems is often indistinguishable from real human voices.

**Di-phone synthesis** uses the minimal speech database containing all the diphones (sound-to-sound transitions) occurring in the language. If you have recorded two phones, then the segment from the middle of first phone to the middle of the second phone will give you the Diphone of these two adjacent phones. Co-articulation can thus be captured and a TTS build in this approach is intelligible and prosody can be incorporated in it more easily. Prosody modeling plays a vital role in this technique. For concatenation of units and prosodic modification, speech models such as Linear Prediction (LP) can be used. Other synthesis techniques such as Time Domain Pitch Synchronous Overlap Add method (TD-PSOLA) can produce reasonable quality speech. Diphone Synthesis technique uses extensive signal processing, which leads to unavoidable degradation of the synthesized speech signal.

## **3.1.3 Articulatory synthesis**

It has been a synthesis method mostly of academic interest until recently. It is based on computational models of the human vocal tract and the articulation processes occurring there. Few of these models are currently sufficiently advanced or computationally efficient to be used in commercial speech synthesis systems.

Articulatory synthesis [11] tries to model the human vocal organs as perfectly as possible, so it is potentially the most satisfying method to produce high-quality synthetic speech. On the other hand, it is also one of the most difficult methods to implement and the computational load is also considerably higher than with other common methods thus, it has received less attention than other synthesis methods and has not yet achieved the same level of success.

Articulatory synthesis typically involves models of the human articulators and vocal cords. The articulators are usually modeled with a set of area functions between glottis and mouth. The first articulatory model was based on a table of vocal tract area functions from larynx to lips for each phonetic segment.

For rule based synthesis the articulatory control parameters may be for example lip aperture, lip protrusion, tongue tip height, tongue tip position, tongue height, tongue position and velic aperture. Phonatory or excitation parameters may be glottal aperture, cord tension, and lung pressure. When speaking, the vocal tract muscles cause articulators to move and change shape of the vocal tract which causes different sounds.

The articulators are usually modeled with a set of area functions of the vocal tract between glottis and mouth. The main constraint with this method is that the parts of the human vocal tract like tongue are so complicated that it is almost impossible to model them precisely.

## **3.1.4 HMM-based synthesis**

It is a synthesis method based on Hidden Markov Models (HMMs). In this system, speech frequency spectrum (vocal tract), Fundamental frequency (vocal source), and duration (prosody) are modeled simultaneously by HMMs. Speech waveforms are generated from HMMs themselves based on Maximum likelihood criterion.

## **3.1.5 Hybrid synthesis**

This uses the aspects of formant and concatenative synthesis to minimize the acoustic glitches when speech segments are concatenated.

## **3.2 Tools and Technologies**

**Festival [12, 13]** is a generic speech synthesis tool free software multi-lingual speech synthesis workbench that runs on multiple-platforms offering black box text to speech, as well as an open architecture for research in speech synthesis. It is written in C++ and it is built at University of Edinburgh in the late 90's. The development of FESTIVAL based on Nepali Text to speech synthesis system within Festival we can identify three basic parts of the TTS process:

* **Text analysis:**

From raw text to identified words and basic utterances.

* **Linguistic analysis:**

Finding pronunciations of the words and assigning prosodic structure to them: phrasing, intonation and durations and

* **Waveform generation:**

From a fully specified form (pronunciation and prosody) generate a waveform.

*Fig 3.1: System Overview of TTS*

Pure word sequence

Sequence of phones

**Text Analysis**

* Tokenisation
* Text Normalization

**Linguistic Analysis**

* Phonology
* Letter to sound rule
* POS tagging

**Waveform Synthesis**

**Speech as Output**



**Prosody Generation**

* Duration

Input

Prosodic parameter provided to each phone

1. **Text Analysis**

The Text Analysis part is preprocessing part which analyze the input text and organize into manageable list of words. It consists of numbers, currencies (RS, $, etc.), abbreviations (USA, UK), acronyms (Dr., Mr., Ms.) and idiomatic and transforms them into full text when needed. The preprocessing steps involves text normalization, tokenization and categorization. Text Normalization is the process of converting Non Standard word to Standard word. Nonstandard word here refers to Non Standard Words here refers to the Abbreviated words, telephone numbers, money amount, dates etc. Changing the Non Standard words into the Standard one refers to identifying the full form of abbreviated words, identifying that the given numbers are phone numbers, identifying the given numbers are amount of money, identifying that given cardinal numbers are dates. Identification of tokens involves a high degree of ambiguity. For e.g., '535263' could be referred as five three five two six three as a phone number or as five lakhs thirty five thousands two hundred and sixty three as a cardinal number. Tokenization is done typically based on white spaces regarding the Nepali Texts as there will no space in the Non Standard word so the tokens will be separated using regular expression which is space. Whitespace (space, tab, newline, and carriage return) can be viewed as separators.

Text Categorization is done once the tokens have been identified from the input text i.e. the category of each tokens needs to be identified. Identification of token category involves of high degree of ambiguity. For example '१६५४' could be type of 'year' or a 'cardinal number'. This kind of disambiguation can be handled by set of rules depending upon the context of the numbers developed by developing team itself. Here tags is used to determine whether a certain number is a year or a currency or a floating point number or etc. The technique for categorizing percentages is that a certain number is followed up by a tag ' % '. So if the following tag is found in front of any number then the number is categorized as percentage. For example: १६%.

1. **Linguistic Analysis**

These word sequences will now pass through a linguistic analysis module. Phonetic Analysis converts the orthographical symbols into phonological ones using a phonetic alphabet. Basically known as “grapheme-to-phoneme” conversion. Letters can be transformed into phonemes. Phone is the smallest sound unit. A collection of phones that constitute minimal distinctive phonetic units are called Phoneme. A phone set needs to be developed in order to incorporate all the sound units in a language. Phones are the basic units of sound that make up speech, while phones are associated with the letters of the written language, the phone may be different depending upon the context. Phoneme is a unit of speech sound that does not have any meaning by itself but when we link phonemes together, words are created. The Letter-to-sound sub module takes the word sequences and converts the words sequences to its corresponding phone sequences referring to the spelling of the word.

1. **Prosody Generation**

After the pronunciation has been determine then prosody is generated. The degree of naturalness of the TTS depends on the prosodic factors like intonation modeling, phrasing modelling, amplitude and duration modelling (including the duration of sound and the duration of pauses, which determine the length of the syllable and tempos of the speech). The concept of prosody is the combination of stress pattern, rhythm and intonation in a speech. The prosodic modeling describes the speaker’s emotion.

1. **Waveform synthesis**

After getting sequences of units (each unit correspond to a wave file). These sequences of wave file can be played or stored into a wave file further where speech is listen as output.

The development of Nepali TTS system based on Festival’s Free TTS is basically divided into three phases. Phase-I: Database creation, Phase-II: Text Analysis, Phase-III: synthesis. The first two phases are language dependent while the third phase is language independent. The entire process is shown in fig 3.2.

Nepali Text

Text Normalization

Text Normalization

Linguistic representation

Prosody and intonation information

Exception Word List

Generate Tokens

Concatenate speech segments

Speech

Normalized Nepali Text

Phonemes

List of Phonemes

Sequence of Tokens

*Fig 3.2: Basic Flow Diagram of Nepali TTS synthesis System*

A paragraph of Nepali text in Unicode is given to the system as input. A graphical user interface (GUI) capable of reading Unicode Devanagari is provided for users to write texts. In the TTS system, input text should be normalized before synthesizing. Text normalization (or Tokenization, preprocessing) converts things like numbers and abbreviations into their written out word equivalents. Tokenization provide meaning to those text in the form of abbreviation, numbers etc. The Speech Database is the repository of all the smallest unit of speech called phones or syllables.

# **Chapter 4**

# **Implementation and Discussion**

A TTS system has text fed as input to generate synthetic speech as output for the corresponding text through audio device to create sound and making good use of FreeTTS tool and Java as described below.

The Nepali text is first input into the system. After the text is fed to the TTS system the text pre-processing is done i.e. the input text is tokenized and for each tokens, it is first matched and replaced with its english representation of the input nepali text in the database dictionary. And for the words not found in the dictionary database, undergoes NLP. The text apart from the database is normalized, syllabified according to algorithm that is implemented and after that the syllables of the text is phonetically represented by the English text for each tokens and the tokens are passed to the DSP component to produce the sound of the words that is syllabalized and for other processing in the sound of the input text.

         **Text Processing**

The text input will be in Nepali characters or basically in Unicode characters.eg.”“मेरो देश नेपाल हो।“”. The text is input in Nepali using keyboard that follows the Nepali words and text. The text processing module consists of preprocessing and syllabication module. The preprocessed text is further on the syllabication module.

         **Syllabication**

In this approach, the syllabication algorithm breaks a word such that there are minimum numbers of breaks in the word, as minimum number of joins will have fewer artifacts. The algorithm dynamically looks for polysyllable units making up the word, cross checks the database for availability of units, and then breaks the word accordingly. If polysyllable units are not available, then algorithm naturally picks up smaller units. This mean, if database is populated with all available phones of language along with syllable units, algorithm falls back on phones if bigger units are not available. A syllable types are: V, VC, CV, VCC, CVC, CCVC and CVCC etc. where V and C represent vowel and consonant respectively that are used for languages. There are twelve vowel found in Devanagari language the 12 Devanagari vowels. Devanagari script also has about 36 consonants.

## **4.1 Phone Set**

Nepali phone set contains 11 vowels, 37 consonants and 21 diphthongs. We have stored these phone set into the speech dictionary to get pronounced by the system which are as below.

|  |  |
| --- | --- |
| **Vowel Alphabets (Swar Barna)** | अ, आ, इ, ई, उ, ऊ, ए, ऐ, ओ, औ, अं, |
| **Consonant Alphabets (Byanjan Barna)** | क, ख, ग, घ, ङ, च, छ, ज, झ, ञ, ट, ठ, ड, ढ, ण, त, थ, द, ध, न, प, फ, ब, भ, म, य, र, ल, व, श, ष , स, ह, क्ष, त्र, ज्ञ |
| **Half consonants** | क्, ख्, ग्, घ्, ङ्, च्, छ्, ज्, झ्, ट्, ठ्, ड्, त्, थ्, द्, ध्, न्, प्, फ्, भ्, म्, य्, र्, ल्, व्, त्र् |

*Table 4.1: List of Nepali Phone Set*

## **4.2 Software Tools Used**

The Software used for the development of TTS system is basically only the Java Netbeans IDE. Making use of the Java GUI tools for the development of User Interactive Interface for taking the input or reading the Nepali Text as Unicode and processing of the text is possible using Java Speech API i.e. FreeTTS, a tool for generating Speech as Output.

1. **JSAPI (Java Speech Application Programming Interface)**

JSAPI is a programming interface provided by Java for speech based applications under Java. JSAPI supports both speech synthesis for text-to-speech generation and speech recognition for recognizing voice. There are many implementation of JSAPI such as FreeTTS, an open source implementation, for text-to-speech conversion. However, it only provides speech synthesis engines. It doesn’t support speech recognition. For this it requires another implementation of JSAPI called Sphinx 4. In addition to this, for naturalness and prosody like of real speech FreeTTS does depend on JSGF and JSML that would increase overhead for the project.

Free TTS [19] is a speech synthesis system written entirely in Java programming language. It is based upon Flite (a small run-time speech synthesis engine developed at Carnegie Mellon University). Flite is derived from Festival Speech Synthesis System from University of Edinburgh and FestVox project from Carnegie Mellon University. Free TTS now has the ability to import voice data from FestVox (US English only). With this, user can record their own voice using the FestVox tools, and then turn the resulting data into a Free TTS voice.

The primary reason behind using JSAPI is because of its support for both speech synthesis and recognition engines. Though speech recognition is not required in our project but its support could be used for extension within the project in future.

## **4.3 Flow Chart**

Input Nepali Text Sn

Tokenization

Replace text with Phonetically Equivalent English word From Dictionary (Wi)

Speech Output

Yes

Text analysis: Normalization

Syllabification

क् + अ=क क् + आ =का

क् + आ + म् + अ = काम

Phonetically equivalent English representation of Nepali phonemes (Wi)

Input Se to DSP Component

FREE TTS speech engine

No

i ++

For each token ti

Map ti from Dictionary D

Append Wi to Se i.e. Se+=Wi

*Fig 4.1: Flowchart of Nepali TTS*

The system takes the Devanagari input and tokenises each word in the input. Then, the tokens are checked and replaced in the database.And if not found in the dictionary, the text is goes NLP process where it is normalized i.e. it converts the non-standard word into standard words as the sentence may contain the abbreviation, numbers, and it needs to be converted into the words done by the normalization process.

It then undergoes for syllabification (i.e. no of vowel with half consonant used in the formation of word like क् + आ + म् + अ = काम. Here अ and आ is a vowel with half consonant क् and म् to form a word काम).

And phonetically equivalent English representation of Nepali phonemes is done (i.e. काम = kaama) for each tokens. Then those sequence of tokens are put into the FreeTTS[19] speech engine from where the speech output is generated.

## **4.4 Algorithm**

1. **Algorithm 1: Word Mapping from Dictionary,**

*Input:* Tokens Ti, where i = 1,2,3,…n

*Output:* English Words (Wi).

**Begin,**

Step 1: Map Ti to Wi where Wi є D, D is the set of words Wi in

Dictionary

Step 2: If successful

Return Wi

Else

Return Ti

**End.**

1. **Algorithm 2: Normalization of Tokens**

*Input:* Token Ti from Input String

*Output:* Normalized Word (Wi)

**Begin,**

Step 1: Wi 🡨 Ti

Step 2: For Every Character/Sub Token Sti in Token Wi, Check whether

if the Sti is Numeric or Abbreviation,

Step 3: If so, Map Sti to its corresponding normalized representation

and update Wi.

Step 4: Repeat 2 and 3 until mapping is complete for all numbers and

abbreviation in the list.

Step 5: Return Wi, which is normalized

**End.**

1. **Algorithm 3: Syllablification of Tokens**

*Input:* Token Ti from Input String

*Output:* Syllablified Word (Wi)

**Begin,**

Step 1: Wi 🡨 Ti

Step 2: For Every Character/Sub Token Sti in Token Wi, Map Sti to its corresponding syllabie representation and update Wi.

Step 3: Repeat 2 until mapping is complete.

Step 4: Return Wi, which is syllabified

**End.**

1. **Algorithm 4: Phonetic Representation of Tokens**

*Input:* Token Ti from Input String

*Output:* Phonetically equivalent Word (Di) for Token Ti

**Begin,**

Step 1: Di 🡨 Ti

Step 2: For Every Character/Sub Token Sti in Token Di, Map Sti to its corresponding phonetic representation and update Di.

Step 3: Repeat 2 until mapping is complete.

Step 4: Return Di, where Di is Phonetical equivalent of Ti

**End.**

## **System Flow Chart**

1. **Use Case Diagram**

Normalize text

Synthesiser

Text Analyser

Java FrontEnd

Open and read text file

Syllabification and Phoneme Mapping

Speech Output

Preprocess text

Supply Input

*Fig 4.2: Use Case Diagram*

1. **UML Class Diagram**

Interface of Nepali TTS

Main ()

Normalization

Normalize ( )

Syllabification

Syllablify ()

Phonetically

Phonetize ()

TTS Engine

Speak ()

TTS Engine

Save ()

*Fig 4.3: UML Class Diagram*

## **4.6 Natural Language Processing**

Natural Language Processing (NLP)[14, 15] is a field of computer sciences and linguistics concerned with the interactions between computers and human (natural) languages. It converts the information from computer database into readable human language. The natural language processor is what knows the rules of English grammar and word formation (morphology). The natural language processor is able to determine the part of speech of each word in the text and thus to determine its pronunciation. More precisely, here's what the natural language processor does:

* Expands the abbreviations, etc. to full text according to a dictionary.
* Determines all possible parts of speech for each word, according to its spelling (morphological analysis).
* Considers the words in context, which allows it to narrow down and determine the most probable part of speech of a word (contextual analysis).
* Translates the incoming text into a phonetic language, which specifies exactly how each word is to be pronounced (Letter-To-Sound (LTS) module).
* Assigns a “neutral” prosody based on division of the sentence into phrases.

# **Chapter 5**

# **Analysis and Evaluation**

## **Analysis**

Nepali TTS is being developed using the framework of Free TTS which is a part of Festival Speech Synthesis System developed by University of Edinburgh. This is a free software which supports multi-lingual speech synthesis and has an open architecture for research in this field. Free TTS is a speech synthesis system written entirely in Java programming language. It is based upon Flite (a small run-time speech synthesis engine developed at Carnegie Mellon University). Flite is derived from Festival Speech Synthesis System from University of Edinburgh and FestVox project from Carnegie Mellon University. Free TTS now has the ability to import voice data from FestVox (US English only). With this, user can record their own voice using the FestVox tools, and then turn the resulting data into a Free TTS voice. For the project Free TTS is used as the DSP component for the Development of the TTS system.

Festival is a generic speech synthesis tool free software multi-lingual speech synthesis workbench that runs on multiple-platforms offering black box text to speech, as well as an open architecture for research in speech synthesis. It is designed as a component of large speech technology system. It is written in C++ scheme.

### **Theoretical Analysis**

The basic units of the writing system are referred to as *Aksharas.* The properties of Aksharas are as follows:

* An Aksharas is an orthographic representation of speech sound.
* Aksharas are syllabic in nature.
* The typical forms Aksharas are V, CV, CCV and CCCV, thus have a generalized form of C\*V.

As known, Nepali script is rich in its vocabulary. The script uses combination of phonemes for its existence. Such as

क् + अ = क, क् + आ = का

To handle this problem the characters with their respective combination counts are stored in the speech or word database.

## **Evaluation**

**Mean Opinion Score(MOS)**

Mean Opinion Score(MOS) [20] is the well known subjective scoring method used to measure the naturalness of the system. For Evaluation Listeners are asked to rate the speech quality of the system, usually synthesizing the same set of sentences. Typically subjects are asked to rate the naturalness of the synthesis on a scale of 1-3, where 1 is poor, 2 is average and 3 is natural or gives correct pronunciation. All results are summed, and a mean score between 1 and 3 is derived, which is meant to represent the oral naturalness rating for the system. The formula of the MOS is defined as

1. **Weight** = Words in category \* Rate
2. **Total Weight** = Total Words \* Maximum Rate
3. **Accuracy** = N/M\*100%

Where, N= total no of word in each category

M= total no of word used in test case

1. **Overall Performance** =

The evaluation is done on the basis of :

1. Intelligibility (how much of the spoken word can be understoond)
2. Naturalness or Pronunciation (how close to human speech does the output of the TTS system sound)

Test cases has been done to know whether the system has natural sounding or not as well as does it, sound closer to the real sound as human speak or not.

The complete result of the Evaluated words is mentioned in Appendix A.

The sample of the method of Evaluation of words is detailed in Appendix B.

### **Evaluation of Vowel, Consonants, and Numeric**

The Nepali language has 36 consonants and 11 vowels and 10 numeric charcters for which evaluation was done and the results as tabulated as below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N** | **Categories** | **Perfect Sounding** | **Average Sounding** | **Poor sounding** |
| **1.** | **Number** | ०, १, २, ५, ६, ७, ८ | ३ | ४, ९ |
| **2.** | **Vowel** | आ इ ई उ उु ए ऐ ओ औ | अ | अं |
| **3.** | **Consonant** | क ख ग च छ झ ट ड त थ द ध न प फ भ म य र ल व स ष श ह त्र | घ ढ ठ ण ज ब क्ष | ङ ञ ज्ञ |

*Table 5.1: Evaluation of numbers, vowel and consonant used in test case.*

Here, the test case has been done to determine the naturalness or accurate pronunciation of the numeric pronunciation in our TTS system. The work has been done by dividing the task into three categories i.e. perfect, average, poor sounding. According to the evaluation, the numbers characters like ०, १, २, ५, ६, ७, ८ are pronunced perfectly, the average sounding number was ३ and poor sounding numbers includes ४, ९. After the calculation, the accuracy was obtained and the accuracy for the perfect sounding is 70%, average sounding is 10%, and poor sounding is 20%. In this test case, rating is done for each categories.i.e. for perfect sounding rate is given as 3, for average rate is given 2 and for poor rate is given 1. From which the weight is calculated for each categories to obtain the overall performance of number in our TTS system and its overall performance accuracy is 83.33% and detailed in *Table 5.2*.

|  |  |  |  |
| --- | --- | --- | --- |
| **Categories** | **Perfect Sounding** | **Average Sounding** | **Poor sounding** |
| **Numeric (0-9)** | 7 | 1 | 2 |
| **Accuracy Percentage** | 70 | 10 | 20 |
| **Overall Accuracy** | 83.33% | | |

*Table 5.2: Evaluation Table for numeric keywords*

The perfectly pronounced vowel are आ इ ई उ उु ए ऐ ओ औ and its accuracy is 81.81%, the average sounding vowel is अ and its accuracy is 9.09%, the poor sounding vowel is अं and its accuracy is 9.09. After calculating the weight according to the rating method the overall performance accuracy is obtained i.e.90.90% and detailed in *Table 5.3*.

|  |  |  |  |
| --- | --- | --- | --- |
| **Categories** | **Perfect Sounding** | **Average Sounding** | **Poor sounding** |
| **Vowels** | 9 | 1 | 1 |
| **Accuracy Percentage** | 81.81 | 9.09 | 9.09 |
| **Overall Accuracy** | 90.90% | | |

*Table 5.3: Evaluation Table for Vowels keywords*

Also, the individual character or consonant that is perfectly pronounced are क ख ग च छ झ ट ड त थ द ध न प फ भ म य र ल व स ष श ह त्र and its accuracy is 72.97%, average sounding consonants are घ ढ ठ ण ज ब क्ष and its accuracy is 18.91%, poor sounding consonants are ङ ञ ज्ञ and its accuracy is 8.10%. Then weight is calculated according to the rating and then overall performance accuracy is obtained which is 88.28% and detailed in *Table 5.4*.

|  |  |  |  |
| --- | --- | --- | --- |
| **Categories** | **Perfect Sounding** | **Average Sounding** | **Poor sounding** |
| **Consonant** | 26 | 7 | 3 |
| **Accuracy Percentage** | 72.97 | 18.91 | 8.10 |
| **Overall Accuracy** | 88.28% | | |

*Table 5.4: Evaluation Table for Consonant keywords*

The obtained data after analysis and evaluation is represented in the bar chart as shown in *Fig 5.1*.

*Fig 5.1: Evaluation Chart for Vowel, Consonants and Numeric*

### **5.2.2 Evaluation for Individual words**

For the evaluation of the individual words, the total words tested during the test case were 2000 out of which total natural sounding words are 1318, average sounding words are 257, and poor sounding words are 425. After analysis, the system had accuracy of 66% well pronunciation and 12.85% average and 21.25% poor pronunciation and the weight of the overall performance was found to be 81.56%.

|  |  |  |  |
| --- | --- | --- | --- |
| **Individual words** | **Perfect Sounding (3)** | **Average Sounding (2)** | **Poor Sounding (1)** |
| **Words used in test cases** | 1318 | 257 | 425 |
| **Accuracy of total words in percentile** | 65.9 | 12.85 | 21.25 |

*Table 5.5: Evaluation Table for Individual keywords*

**Pie Chart Represntation of the Accuracy of the System**

*Fig 5.2: Pie Chart Representation of Accuracy of Pronunciation*

**Bar Chart Representaiton of Evaluation of words**

*Fig 5.3: Evaluation of Words*

**Calculation**

Total Weight = Total Words \* Rate

Accuracy = N/M\*100%

**Where**, N= total no of word in each category

M= total no of word used in test case

Using the above Formula and By Mean Opinion Score(MOS),

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Individual Words** | **Total Words** | **Natural Sounding (3)** | **Average Sounding (2)** | **Poor Sounding (1)** |
| **Weight of words in each categories** | 6000 | 3954 | 514 | 425 |

*Table 5.6: Weight of Words in each Categories*

**Overall Performance**

The overall performance defines the accuracy of the pronunciation of the system and gives the performance of the system.

Overall Performance =

=

= 81.525

|  |  |
| --- | --- |
| **Overall Performance for the System** | |
| **Total words tested** | 2000 |
| **Weighted Words Value** | 6000 |
| **Overall Porformance** | 81.52 |

*Table 5.7: Evaluation of Total Words for Accuracy*

Overall performance of the Nepali TTS system was calculated to be 81.25% of the developed Nepali Text to Speech System.

# **Chapter 6**

# **Conclusion and Future Work**

## **6.1 Conclusion**

As we know that there exist TTS (Text to Speech) system for different languages. Nepali TTS system is an application for the Nepali language that accept the input as a text and then produce the audio speech. TTS system is an application for Nepali language that converts normal language text into speech i.e. the system that render symbolic linguistic representation like phonetic transcriptions into speech.

NeLRaLEC or Bhashasanchar’s Nepali TTS project was developed as the first Nepali TTS developed under the framework of Festival Speech Synthesis System based on Concatenative Approach for synthesis.

The method for developing a TTS system requires text processing, tokenization, normalization, linguistic representation, prosody, Wave form generation to give audible sound for the words or text.

Hence, Nepali TTS system is developed; that produce the audio speech by describing the various steps involved in the Natural Language Processing based on framework of Free TTS; a method of formant approach giving more emphasis on the NLP process rather than the tasks of the DSP process, making complete use of JAVA, Java Components as well as its APIs.

On the basis of Analysis and Evaluation, the developed System has Overall Performance of 81.52% and the accuracy of pronunciation of words is found to be 66% natural, 13% average and 21% poor pronunciation.

In the current system, linguistic feature such as intonation and prosody has not been implemented. Further enhancement of the system incorporating these features, still remains.

However the system produces flat speech for any inputted Nepali text, which doesn’t sounds as natural as spoken by a human. The system can be extended to include more features such as more emotions, improved tokenization and use of minimal database.

## **Limitation**

* It doesnot read input for date and time numbers to text i.e. for example २०५० is given as an input to the system then it would speak it as “दुई सुन्य पाँच सुन्य” instead of “दुई हजार पचास”.
* It doesn’t sound as natural as human speak.
* No natural Nepali pronunciation.

## **6.3 Future Work**

* The system can be extended to include more features such as more emotions (orfeelings like anger, happiness and sadness).
* Improvement of the smoothness of the sound.
* Inclusion of prosody and the naturalization of the voices like human expressions.
* Reading of special cases like date and numbers.
* Inclusion of different kinds of voices (i.e. male and female) and graphical faces.
* Controlling the reading speed i.e. duration between the words.
* Handling exception words that doesnot sound natural.

# **References**

[1] Dutoit T., “An Introduction to Text-To-Speech Synthesis”**¸** *Kluwer Academic Publishers***,** 1996.

[2] Allen J., Hunnicut S., Klatt D., “From Text To Speech, The MITTALK System”, *Cambridge University Press*, 1987.

[3] J.Sangeetha,S.Jothilakshmi , S. Sindhuja , V. Ramalingam ,Text to Speech synthesis system for Tamil, International Conferenceon Information Systems and Computing (ICISC-2013),India.

[4] K. Kamble and R. Kagalkar,A Review:Translation of Text toSpeech Conversion for Hindi language, International Journal of Science and Research (IJSR) Volume 3 Issue 11, November,2014.

[5] Van Santen, Jan P. H.; Sproat, Richard W.; Olive, Joseph P.; Hirschberg, Julia (1997). Progress in Speech Synthesis. Springer. ISBN 0-387-94701-9.

[6]NepaliTTS.Online:http://www.bhashasanchar.org/pdfs/NepaliTTS\_%20manual.pdf. Access Date: 14th October, 2015.

[7] Hariharan, R. [Online]. Available: http://dhvani.sourceforge.net/. Accessed on 20 February 2012.

[8] Speech Synthesis. Online: http://en.wikipedia/wiki/Speech\_Synthesis. Access Date: 4th November, 2015.

[9] D. Klatt, “Software for a cascade/parallel formant synthesizer,” Journal of the Acous-tical Society of America, vol. 67, pp. 971–995, 1980.

[10] D. Jurafsky and J.H. Martin, Speech and Language Processing. Pearson Education, 2000.

[11] P. Rubin, T. Baer and P. Mermelstein, “An articulatory synthesizer for perceptual research,” Journal of the Acoustical Society of America, vol. 70, pp. 321–328, 1981.

[12] Black A., Taylor P., Caley R. (1999) The Festival Speech Synthesis System Documentation v1.4. (http://www.cstr.ed.ac.uk/projects/festival/manual/).

[13] Building Synthetic Voices, by Alan W Black and Kevin A. Lenzo, For FestVox 2.0 Edition 1999-2003 by Alan W Black & Kevin A. Lenzo

[14] Dutoit T., “An Introduction to Text-To-Speech Synthesis”**¸** *Kluwer Academic Publishers***,** 1997.

[15] D. Jurafsky and H. James, “Speech and language processing an introduction to natural language processing, computational linguistics, and speech,” 2000.

[16] [David Öhlin] Rolf Carlson "Data-driven formant synthesis" Proceedings, FONETIK 2004, Dept. of Linguistics, Stockholm University.

[17] A. Trilla, Natural Language Processing techniques in Text-To-Speech synthesis and Automatic Speech Recognition. Departament de Tecnologies Media Enginyeria i Arquitectura La Salle (Universitat Ramon Llull), Barcelona, Spain atrilla@salle.url.edu, 2009

[18] “Stephen Hawking - Home.” [Online]. Available: http://www.hawking.org.uk/. [Accessed: 28-Feb-2016].

[19] Free TTS online: http://freetts.sourceforge.net/docs/index.php [Accessed: 28-Oct-2015].

[20] ftp://ftp.scansoft.com/products/realspeak/wp\_RS\_TTS\_EvalTTS.pdf

Appendix

Appendix A: List of all the words used in test cases

1. **Average Sounding**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| किर्तिमान | कता | काट्छ | कम्पनी | कुखुरा | काभ्रे | कैदी | काल | काठ |
| कपिलवस्तु | केश | कोसा | कत्री | कसै | करार | क्यासिनो | कथाबस्तु | कल्पना |
| कालिमाटी | कला | कलेज | कैलाली | कलश | कतै | कती | कस्ती | कस्ता |
| कपूरधारा | कलेजो | कमिला | किरण | कपुर | कपडा | कब्जा | काठमाडौं | कालिनदी |
| कन्यादान | कंचनपुर | खाउन | खाई | खोटाङ | खुलेमा | गैंडा | घुम्ती | घटना |
| चन्द्रबहादुर | च्यामासिङ | चाहियो | चाहिन्छ | च्याङ्ग्रा | चित्रकार | चुङ्गी | चन्द्रमुखी | चिम्टा |
| चियापत्ती | छाना | छाया | छाता | छल | छाउनी | छोरी | छोरा | छत |
| जानकारी | जनमत | जंगली | जोड्न | जान्छौ | जमरा | जेठान | जेठानी | जेठो |
| जनप्रतिनिधी | जेठी | जयन्ती | ज्यादै | जनजाती | जनमोर्चा | झलनाथ | झरना | टहरा |
| ठमठम | ठोक | ठुलो | ठेकी | ठुलै | ठुली | डेउडा | डाक्टर | ढक्की |
| तालिका | तामाङ | तानेर | ताल्चा | ताली | ताजा | तालिम | तोकेर | तल |
| तिमील्सिना | तात्पर्य | तल्लो | तातो | तान्त्रिक | थैली | थप्प | थकाई | थाली |
| दारसानीक | दर्ता | दाजु | दाल | दाजुभाइ | दार्जिलिङ | दिम्पल | दक्षिण | दुर |
| दयनिय | दिप्ती | दिएन | दाँया | दुर्गन्द | दाउरा | दङ्ग | धातु | धैर्य |
| धमाधम | धागो | धन्दा | ध्यान | धुन्न | धम्की | नायक | नेवारी | नमक |
| नामाकरन | निस्चल | नदिन | न्यानो | निसानी | नपाएर | नपाई | नमिठो | निरन्कुश |
| निजामती | निमन्त्रण | पारी | परेवा | पानी | पवन | पङ्खा | पार्वती | पोखरा |
| प्याज | प्रतिकुल | प्रतिफल | पानीपुरी | परलोक | पालुङ | पाहुना | पधमपुर | फागु |
| फ्यान | फसाद | फागुन | फैसला | फेरी | बालुवा | बालिका | बाहिर | बाला |
| बाहुन | बाल | बधाई | बगेर | बोहोता | बाहुन | बारी | भाडा | भालु |
| भन्छन | भाङ | भन्दा | भेरी | भेटी | भेटेर | भोग्न | भोग्नु | भेट |
| भुतपुर्व | भुल | भिन्न | मालिक | मामिला | म्याथ | मान्छे | मदन | मधुरो |
| माहापुरुष | मिहिनेत | मन्जरी | मार्ग | योजनाबद्ध | यती | राख्यो | रमिता | राजिव |
| रामायाण | राती | राप | लाटी | लौका | लागत | लडाईं | लाउन | लेखा |
| लेख्नुभयो | लेखी | लेख | वारीपरी | वरिपरी | वकिल | वार्ड | साहस | सारी |
| समुदायिक | सार्थक | सभामुख | साथसाथै | सावित्री | समिती | सरस्वती | संसार | सान |
| समाजसेवी | शासन | श्वेत | शैशव | शान्त | शान्ता | शाक्य | षडशी | हेरे |
| हावाहुरी | हेर | हाट | होइन | हूल | हात | होलिन् | हली | क्षय |
| क्षेपक | क्षत | त्रियासी | त्रिपाठी | त्रिनेत्र |  |  |  |  |

1. **Poor Sounding**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| कोशीस | कालु | केही | कहिले | कथन | कारण | कोमल | कागती | कैद |
| क्यालोरी | कौसिला | कौसी | कम्मर | काट्ने | कतिपय | कामी | काले | केरा |
| कारखाना | कमाइ | काला | कैलाश | कैलेश | कहिले | कापी | काफ्ले | काध |
| कल्यान | कामना | कालो | काकी | काजी | काका | कुकुर | कथा | कसैलाई |
| काहानी | खाएछ | खाएउ | खाँदैछ | खुशी | खाए | खासै | खरायो | खाने |
| खानेछ | ख्यालै | खुले | ख्याल | खोस्ने | खोज्दा | खेल्दै | खैजडी | खेले |
| खरानी | खासा | खान | खानुपर्छ | खानपिन | खाईस | खन्चुवा | खसेछ | खाने |
| गणेश | गणेशमान | गन | ग्यास | गर्नुपर्छ | गम्छा | गगन | गर्नेछौ | गर्ने |
| गर्दछौ | गद्दी | गफ | घर | घाऊ | घाम | घेरा | चरा | चाहे |
| चाहना | चाबेल | चमेरा | चालक | चाउचाउ | चेला | चाहिन | चाहिन्न | चाहिने |
| च्यातिन्छ | चापी | चिलायो | चलाउने | चोरी | चिट्ठा | चुरोट | च्याट | चालु |
| च्यात्तियो | चुस्ने | चपाउने | चोर | चाम्रो | चामल | चरेस | चेली | चर्चित |
| चराचुरुङ्गी | चल्दै | चारै | चलिरहे | चलाख | चक्चके | चाला | चिकेन | चले |
| चालामाला | चल्ला | चाहने | छक्क | छहरा | छाती | छप्पा | छन | छेउ |
| छोराछोरी | छाडे | छाउरो | छम | छोडि | छोपे | छोडे | छोड्ने | छेउछाऊ |
| छम्छ्मी | जाली | जानेछन् | जान्न | जाने | जादु | जानेछन् | ज्यान | ज्वाँई |
| जेटप्लेन | जाभा | जथाभाबी | जागिर | जारी | जानुपर्ने | जड | जहिले | जंगल |
| जरिवाना | झन्डै | झ्याली | झाडु | टाउको | टाका | ठाडे | ठाउँ | ठेली |
| ठुले | ठग | डन | डारा | डाडा | ढकनी | ढाक | तारा | तिर्ने |
| तताउने | तारिफ | तोड्फोड | त्यही | त्यस | त्यै | त्यस्बेला | त्यसैगरी | त्यसैले |
| ताकी | त्यो | त्यत्रो | त्यतिखेर | त्यतिबेला | त्यती | त्यस्तै | त्यस्ता | त्यसो |
| त्यसैकारण | त्यसरी | त्यसै | थाल्छन | थाल्छ | थालेछन | थाले | थुम्के | थोपा |
| थालिसके | दाहाल | दैबी | दामन | देखापर्ने | देखापर्छ | दिने | दाँया | देखे |
| देबकोटा | देन | देख्ने | दिनुपर्छ | दुरी | दिने | दुर्भाग्य | दुइचार | दिनुपर्ने |
| देखिन्थ्ये | दारा | धामी | धान | धन | धरान | धार | धक्का | धारा |
| नातिनी | नुहाउने | नाच्छन | नारायण | नागिन | नारा | नाउने | नबुझे | नदेखे |
| नोक्सानी | नचाहिने | निस्चित | नाली | निकाल्ने | निकाले | नेताजी | नेता | नङ |
| नयन | नतिजा | नसक्नु | नसके | नरहे | नरहने | नपर्ने | नपाउने | नमिले |
| नच्ने | नलगी | नलागे | नागिन | नारायणी | नहुने | नजाने | नक्साल | नाती |
| नारायणगड | नदिने | पोखेर | पोखरेल | पोखरी | प्रग्या | पैसा | पारश | परौठा |
| पाउरोटी | पाखा | पुर्णिमा | पुर्ने | पुने | पखेटा | पारा | पंछी | पापी |
| पश्चिम | पाटन | पस्ने | परेछ | पिउने | परीरहे | परेनछ | पकाउने | पर्छ |
| फर्काउने | फुटे | फन | फरक | फोन | फर्के | बुझिन्छ | बाबु | बैनी |
| बालबालिका | बाहेक | बल्ल | बस्यो | बाजे | बानी | बगेर | बारा | बाहेक |
| बिदाई | बिहे | बिर्सेंर | बिसौनी | बकैदा | बैठक | बैदार | बुझे | बुझेर |
| बुझ्ने | बुझाई | बोल्ने | बेपारी | बेपार | बालक | भाग्यो | भन्ने | भाऊ |
| भावनात्मक | भाव | भारतीय | भाग्य | भावना | भन्न | भाबुक | भदौ | भाद्र |
| भएनन | भएछ | भर्ने | भन्थे | भावी | माझ | माग्ने | मणी | मल्ल |
| मालपोत | मान्दै | मिलाउने | मर्छन | मस्त | मरे | मेरी | यसप्रकारले | यसैले |
| रिसाए | राजन | राख्न | राख्दै | रत्न | रोचक | राख्ने | लागिन | लागिन् |
| लाग्दछ | लागे | लगाइसके | लगाउने | लिनुपर्छ | लाग्नेछ | लाने | लैजाने | लोग्ने |
| लोक्साहित्य | ल्याउछ | लाग्छ | ल्याउने | लोकगीत | लाग्यो | लोग्नेमान्छे | विवेक | व्यथा |
| वेवस्तापन | वेवस्ता | विचरा | विदेशी | व्यस्त | व्यक्ती | वन | वैद्ध्य | वैदिक |
| सुब्रिग्य | सानी | सामुन्ने | सौताने | सक्छ | सामन | संकेत | सकिने | सारे |
| साली | सोझो | सेतो | सुके | सँग | सन्चो | षड्कोण | हाल्छ | हैट |
| हिमानी | हान्ने | हेर्दै | हलो | हुने | हुनेछ | हुनुपर्दछ | हुन्छे | हाल |
| क्षण | क्षेम | त्रिकोण | ज्ञात | ज्ञानी | ग्यालिन | ज्ञानु | ज्ञान | ज्ञवाली |
| ग्यालेक्सी | ज्ञाता | ज्ञानकुन्ज |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

1. **Natural Sounding**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| कविता | कसम | कोही | कान | कलम | कार्की | काम्दार | कस्मिर | कोया |
| कवी | काफल | कत्ती | कुरा | कुसुम | काव्य | कोल्गेट | कर्तब्य | कोट |
| कृति | कमला | कुनै | केन्द्रित | कम्जोरी | काव्या | काट्नु | कोइराला | कोटा |
| काल्पनिक | कान्तिपुर | कर्मचारी | कस्तो | कर्तूत | कास्की | कप्टी | कर्म | कोरा |
| कलस | कलर्स | कसो | कत्रो | काम्काज | कम्जोर | किलो | कसरी | कोदो |
| कान्छी | कर्ता | काग | कपाल | काँचो | कुल्फी | किल्ला | कुन | किताब |
| कान्छा | कोलोम्बु | कुल | काइयो | कान्छो | कोकाकोला | किला | कम्तिमा | किन्न |
| कोहिनूर | कल | काम | कमल | काँक्रो | कफि | किरा | केरा | किन्तु |
| क्रान्तिकारी | किनेर | कुटेर | कुर्सी | कुरै | कुराकानी | कुनै | कुन | कुमारी |
| कुलो | क्रोसिया | क्रोध | क्रमश | कहाँ | खानु | खतरा | खोज | खोइ |
| खोल | खोला | खोजी | खराब | खान्छ | खानिखोला | खुबै | खुब | खोइत |
| खाजा | खम्बा | खजुरी | खस्यो | खान्छु | खाँदैनौ | खसी | खोलानाला | खायो |
| खेर | खाल्टो | खाली | खाना | खान्छन | खान्छन् | खायो | खगेन्द्र | खल्ती |
| खनाल | खम्बा | खोज्नु | खोजेर | खोज | खोजी | खिसी | खेती | खेत |
| खेर | खेल्न | खेल्दा | खत्री | खतम | खोज्यो | खोली | खोलेर | खोल्न |
| खोलो | खोसेर | खुकुरी | खुला | खुल्यो | खुबै | खुलेर | खुल्ला | खुब |
| गमला | गजुर | गजुरी | गाई | गोरु | गीत | गाउँ | गाईस | गायक |
| गायिक | गिलास | गण्डकी | गुलाब | गुम्राह | गुम | गम | गुण्डा | गुरु |
| गोपाल | गर्दिन | गर्नुपर्दा | गर्नु | गर्नुहोस् | घिमिरे | चाड | चाडपर्व | चमेली |
| चक्लेट | चरी | चप्पल | चनाचट्पट् | चम्चा | चिया | चिनी | चम्सुर | चना |
| चाहन्छन् | चाहन्छौ | चाँहि | चाहन्छ | चम्पा | चल्नु | चाँहिदैन | चाख | चाकु |
| चक्कु | चप्पा | चलाउनु | चलेर | चलाऊन | चलाइ | चक्र | चकटी | चासो |
| चारा | चैत | चैते | चल्ती | चल्यो | चमत्कार | चनाखो | चन्दा | चन्दन |
| चर्को | चेतन | चुइगम | चौथाई | चौतारो | चौतारी | चौधरी | चरम | चन्द्रिका |
| चन्द्रमान | चन्द्रमा | चन्द्रदत्त | चन्द्रकला | चन्द्र | चुम्बक | चनौली | चुनौती | चेपारो |
| चिकित्सक | चिल | चिप्रा | चमेरो | चोली | चिरन्जिवी | चरन | चिम्सो | चाइना |
| चिनारी | चीन | चीना | चिन्ता | चिन्त्तित | चित्र | चित्त | चोटि | चेतावनी |
| चुलैनिम्तो | चुकुल | चेपाङ | चन्द्रा | चुचुरो | चङ्गा | चन्दा | चीज | चूर |
| चुलो | चिप्लो | चुप | चर्पी | चिउरा | चिसो | चेतना | छाडेर | छाला |
| छोपेर | छर्यो | छिमेक | छिमेकी | छानिन्छ | छान्छन | छुन्नौ | छोडेनन् | छोड्न |
| छोडेर | छुन | छुट्टै | छोटो | छिटो | छबी | छरे | छैन | छट्टु |
| छापाखाना | छेपारो | जुरेली | जन्ती | जाडो | जाल | जान्छन | जजमान | जन्तु |
| जरायो | जुम्रा | जून | जुनकिरी | जुस | जम्प | जहाज | जम्काभेट | जम्बो |
| जस्तोसुकै | ज्यादा | जुरुक्क | जोस | जोडेर | जेजस्तो | जिउ | जनसंख्या | जान |
| जोर | ज्योति | जुनसुकै | जोर्नी | जिप | जेल | जटिल | जरुरी | जमान |
| जुटाउन | जुत्ता | जुम्ला | जोडी | जेला | जतिसुकै | जस्तो | जनकपुर | जमाना |
| जमुना | जप्न | जाओस् | जनता | जस्ता | जबकी | जान | जन्त | जानकी |
| जनअन्दोलन | जलस्रोत | जस्तै | जसरी | जम्मै | जमिन | जफत | जनजीवन | जिरा |
| जिब्रो | जिविका | जता | जताततै | जगत | जग्गा | जसरी | जाउ | जनक |
| झोला | झिकेर | झिकी | झापा | झाप्पड | झुपडी | झुत्तो | झुप्पो | झुटो |
| झुल्यो | झार | झन्डु | झन्डा | टपर | टपरी | टाईम | टाईप | टिका |
| टोली | टिकट | टिक्का | टिक्न | टुप्पी | टुप्पो | ट्रेन | ट्रेड | टुक्रा |
| टोल | टेबुल | टि.भी | टिका | टिकाराम | टोकी | टोक्यो | टोक | टीन |
| ट्रम | टिप्पर | ट्रक | ट्रली | टुनामुना | ट्रन्कल | टेलिकम | टेलिफोन | टप |
| टुक्रा | ट्रोफी | टोली | टिप्पन | टिपेर | टुप्पो | टोपी | टोकेर | ठेगान |
| ठडेउरो | ठेगाना | ठाकुर | डमरु | डेरा | डेरी | डडेलधुरा | डरलाग्दो | डलर |
| डण्डो | डेढ | डन्डी | डन्डा | ढल्न | ढोका | तैपनी | तब | तलाउ |
| तलाब | तलब | तपस्या | तमाखु | तर्फ | तयारी | तरकारी | तर्क | तस्विर |
| तुल्सी | तिब्रगती | तिब्र | तिलक | तिमी | तिल | तोरी | तोक | तोड |
| तिर्नु | तिथी | तिरै | तिर | तिनी | तिनै | तीन | तर | तिम्रो |
| तिम्री | तितौरा | तितो | तपाईं | तार | तन | तसर्थ | तन्त्र | तमाम |
| तत्कालिन | तत्कालै | तेज | तेत्रो | तेल | तब | तस्करी | तथ्य | तर |
| तरुल | तातेर | थाहा | थाहै | थाल | थालिन | थाल्नु | थालियो | थाल्यो |
| थरी | थर | थरीथरी | थोक | थियो | थुन | थुप्रै | थोरै | थिन |
| थी | थिएन | थिच | थिकै | दाई | दिन | दर्शन | दिल | दाम |
| दशै | दसरथ | दस्तुर | दबाब | देह | देख | देखेर | दुध | देखापरे |
| दया | दाउ | देश | दात | दान | दानव | दाना | दाम्पती | दिल्ली |
| दयारा | दबाउन | दही | दैलो | दैनिक | देबी | दश | दल | दलाल |
| दक्षिणपुर्वी | दसा | दर्सक | दर्जन | दर्जा | दक्ष्यता | दरबार | दर | दफा |
| देखिन्छ | देखिन | देखेको | देखी | देखेर | देख्दा | देखाउनु | देखाएको | देखाउन |
| देखिदैन | देखिन्थ्यो | देवील | देवीका | देखियो | देखिन्न | देव | देखिसकेको | देसी |
| दुर्गा | दुर्लब | दुनियाँ | दुई | दिनेश | दिशा | दिव्या | दिव्य | दोस्रो |
| दु:खद | दुखी | दुबइ | दुला | दुलही | दुलो | दिन्छ | दिनहु | दिमाग |
| दियो | दिएन | दिपु | दिर्ग | दिपा | दिपक | दिगो | दिएनन | दिनी |
| दिर्गकालिन | दिवाकर | दिदि | देवेन्द्र | देवीप्रसाद | देबता | देश्भक्त | देश्भक्ती | धान्छन |
| धनी | धार्मिक | धमिलो | धर्ती | धनकुटा | धर्म | धर्मदेव | धर्मग्रन्थ | धेरै |
| धेरैजसो | धोका | धोती | धुलो | ध्रुब | धुन | धुप | नाक | नाच |
| नाकाबन्दी | नागराज | नागरिक | नागरिकता | नाग | नाम | नानी | नाना | नाङो |
| नामावली | नाफा | नदेर | नदी | नभनेर | नभै | नाउ | नाथ | नम्बर |
| नाटक | नशा | नाश | नारी | नगर | नगरौ | नहोला | नहोस् | नहुनु |
| नकारात्मक | नैतीकता | नैतीक | नजर | नजिकै | नहुन | नक्सा | नमुना | नक्ष्यत्र |
| नकटो | नमस्कार | नयाँ | नेपाल | नदिनाला | नपरोस् | नर | नपरे | नराखी |
| नराम्रा | नराम्रो | नरम | नराम्राी | नरबहादुर | नरहरी | नराख | निम्की | नसकेर |
| नसकिने | नौलो | नव | नौ | नत्र | नवलपरासी | नविन | नविना | नेकपा |
| नेपालभारत | निकासी | नेपाल | नेपाली | निद्रा | निर्मल | निर्जला | निशान | निर्मला |
| नेपालगन्ज | निकास | निधो | निकाल्न | निकाली | नक्कली | निकालेर | निकाल | नेवार |
| निकाल्यो | निजिकरन | निच | निजी | निती | निको | निमित्त | निकै | निर |
| निर्दोश | निर्धारित | निर्जिव | निरन्तर | निर्देशक | निर्धारण | निरन्तर | नस्तिक | निरा |
| निर्धक्क | निरौला | निवा | निस्क्यो | निस्कियो | निर्वाचन | नाम | नाला | निस्केर |
| नित्य | नियम | नियत | नियुक्त | निल | नोकरी | नट | नुन | नोकर |
| नोक्सान | नायक | नर्सरी | नर्सिङ | नयादिशा | न्युनतम | पाउजु | परी | प्लस |
| प्रचार | प्रगती | पब्लिक | पापड | प्रसुती | पम्फा | पन्जा | पनेरु | पनीर |
| परोपकार | पौडेल | पत्रचार | पस्यो | पर्‍यो | पत्र | पेसा | पसल | पूजा |
| परीक्षा | प्रसाद | परीचित | पियुन | पर्दैन | परन्तु | पन्थी | प्रेमिका | प्रेम |
| पेट्रोल | पशु | पढ | प्रशान्त | पास | पासा | पलान | प्रलाद | प्रहरी |
| प्रधानमन्त्री | प्रयास | पुतली | पशुपती | पहरो | पुलाऊ | पछेउरा | पराजुली | पागल |
| पशुपतिप्रसाद | पुरी | पहिलो | प्रधान | पुरस्कार | पास्ता | प्रेस्ता | पेसा | पाल्पा |
| पूल | पिंजडा | पवनपुत्र | पाउन | पुत्र | पौष | पूर्ण | पायल | पेपर |
| पेन्सिल | पेन | पेच | पान | पाचक | परेर | फर्सी | फर्मा | फारम |
| फर्किनुस् | फूल | फलफूल | फर्क | फैलियो | फर्केर | फर्किन्छ | फर्कनुस् | फल |
| फुर्सद | फ्रुटी | फुर्ती | फूलबारी | फुकेर | फ्रेम | फोटो | फोरा | फेमिली |
| फिट | फिल्मी | फिल्म | फोका | फोको | फिलिपिन्स | फिर्ता | फेर्न | फेर |
| फौजी | फौज | फर्की | बाटो | बन्चरो | बम | बोल्यो | बबल्स | बाख्रा |
| बुबाआमा | बाँदर | बोक | बोका | बिदा | बिकास | बिक्री | बिहान | बिजुली |
| बसपार्क | बिस | बिच | बिरोध | बिन्दु | बिलौना | बोसो | बोक्सी | ब्रामण |
| बफादार | बोलिन | बोल्नु | बुद्ध | बुबा | बुहारी | बेक्कार | बेफाईदा | बाफल |
| बुढाबुढी | बुढो | बुढी | बमोजिम | बखत | बैशाख | बोल्बम | बजारी | बजार |
| बत्ती | बारुला | बार | बाङ्लादेश | भाउजु | भान्जा | भएको | भाद्र | भनेन |
| भानुभक्त | भात | भागेर | भएपनी | भएर | भगवान | भागवती | भाई | भक्त |
| भन्नुभयो | भोलिपल्ट | भोज | भोको | भोटाहिटी | भुटान | भन | भुमरी | भुइ |
| भोली | भोक | भोगी | भोग | भित्र | भेट्घाट | भेडा | भेद | भयो |
| भौतिक | भट्मास | भरोसा | भण्डारी | भन्छु | भन्छिन | भनाइ | भक्तपुर | भल |
| भौगोलिक | माला | माल | मन्त्री | मुन्नी | मम्मी | माग्न | मैना | माधव |
| माध्यम | मामा | माइजु | मगर | माइती | माइतीघर | मम | मानवता | मानेर |
| मुरली | मुसा | मुलुक | मुक्ती | मुक्त | मुख | मोटो | मोरङ | मोल |
| मोजा | मिल्न | मोही | मिली | मिल्छ | मिल्दो | मेरो | मेरै | मतलब |
| मन्दिर | मनमोहन | मैनाली | मज्दुर | महिमा | महोत्तरी | महिला | महेन्द्र | मागेर |
| मनकामना | मदत | मन्त्र | माथिल्लो | माथि | माता | माया | ममता | माटो |
| मास | मासु | मार्यो | मार्नु | मर्नु | मार्मिक | मर्का | मार | महिना |
| मानिन्थ्यो | यातायात | यात्रा | यात्री | यात्रु | यदी | यहाँ | यस | यसैगरी |
| यसकारण | यसको | यसमा | यसो | यथार्थ | यसरी | यथार्थता | यिनै | यसै |
| युवती | युवराज | युवा | युरोप | युक्ती | युक्त | युग | युद्ध | योजना |
| युद्धबिराम | योग्यता | योग्य | योगी | योगदान | योगा | यिनी | यत्रो | रातो |
| राखी | राम्रा | राम्री | राम्रो | राखियो | रामु | राप्रपा | रहस्य | रात |
| राय | रकेट | रही | रहोस् | रवी | राखम | रित्तो | रिस | राजु |
| रोग | रोगी | रोक | रोकेर | रोमा | रोक्नु | रोटी | रेग्मी | रितु |
| रजनीकान्त | राजा | राजदुत | रानी | रगत | राजकुमार | रजनी | राजगद्दी | राम |
| रहेर | रमेश | रमाइलो | रोजगारी | रुस | रुपैयाँ | रोमन | रुपा | रुन |
| राष्ट्रपती | रहनु | रुन्ची | लाग्दैन | लेर | लाग्दा | लागेन | लागेर | लाग्ला |
| लगायत | लामा | लामो | लाग्नु | लास | लाटो | लाटा | लाइन | लाउनु |
| लगायो | लिखित | लोक | लुम्बिनी | लुकेर | ल्याउनु | लायो | लोकप्रिय | लुगा |
| लाठीचार्ज | लोभ | लोन | लिटर | लियो | लिन | लिन्छ | लिएर | लौरो |
| लब | लहर | लगाइ | लैजान | ललितपुर | लौरी | लक्ष्य | वास्ता | वास्तब |
| वास्तब्मा | वस्तुगत | वेद | विवाह | विन्दु | विनास | विन्ती | विश्व | वस्तु |
| वास्तविक | सुब्रिना | साउन | सर्प | सराप | साबुन | सन्तान | सुन्नु | सानु |
| सुन्नी | शक्ती | सबिता | साधन | साहु | साझा | सारो | साले | साला |
| साध्य | सामाजिक | सामना | सामाग्री | साह्रै | साथी | साना | सहयोग | सिता |
| सामाग्रीहरु | संस्क्रिती | सँस्कार | सामुहिक | सकियो | सकिन्न | सामु | साम्सद | सित |
| सँस्क्रित | सरी | सासु | साप | सरकार | सात | सातो | सात्तो | सास |
| सकेर | सभा | सभ्य | सचिव | सदन | सदस्य | सडक | सब | सबैजसो |
| सधुपयोग | सदस्यता | सधैं | सहायक | समाधान | साबधान | संजीव | सहित | सहज |
| सेबिका | स्वर्ग | सुबेदी | सत्य | सुन्दर | सुमन | सुगन्द | सत्र | स्पर्स |
| सोध्यो | सोही | सकिरा | साहारा | सेबक | सतिदेवी | सती | ससुराल | ससुराली |
| सरु | सस्तो | सरकारी | सन्तोष | सन्देश | समुदाय | सपना | सम्झेर | समिक्षा |
| सम्भावना | समेत | समेट | सम्भब | समाचार | सकिन्न | सकिन्छ | सकेनन् | सक्त |
| समुदायिक | सकेन | सजिलो | सजाय | सैनिक | सहनु | शिशिर | शशी | शब्द |
| शोषण | शास्त्री | शुचि | शिक्षा | शत्रु | शोभा | शास्त्र | शरीर | शिर |
| शर्मा | शाश्वत | शीर्षक | शिरिष | श्वास | शुभ | शोक | शिव | शिष्य |
| षडानन्द | षड्यन्त्र | हल | हट | हुँदैन | हाल्दैन | हुन्छ | हुनुहुन्छ | हुन्छस् |
| हानिकारक | हाल्नु | हाल्दा | हाम्री | हाम्रो | हामी | हानी | हाल्न | हान्छ |
| हान्नु | हजुरआमा | हावा | हुन्थ्यो | हुन्न | हजुरबुबा | हुरी | हुर्केर | हुन्छु |
| हुनुहुन्थ्यो | हुँदैन | हुलाकी | हुकुम | हुलाक | होटेल | होली | हिसाब | हिन्दी |
| हिन्दूधर्म | हिन्दू | हिन्सा | हिम्मत | हुक्का | हर्ष | हमाल | हिमाल | हरी |
| हमराज | हिरा | हित | हेर्यो | हेर्नु | हेर्दा | हेर्नुहोस् | हेर्नुस् | हात्ती |
| हेर्नुहुदैन | हान्नुहुन्छ | क्षेत्रपुर | क्षमा | क्षेपारो | क्षति | क्षमता | क्षेत्री | क्षेत्र |
| त्रैमासिक | त्रिचन्द्र | त्रिसुल | त्रिभुवन | त्रस | त्रिदर्सी | त्रास | त्रिचोक | त्रुती |
| त्रिसुली | त्रिभुज | त्रिफला | त्रियोग |  |  |  |  |  |

# **Appendix B: Sample of Evaluation of Words**



