Word Embedding in Nepali Language using Word2Vec

Bipesh Subedi
Kathmandu University
Dhulikhel, Bagmati, Nepal
bipeshrajsubedi@gmail.com

Prakash Poudyal[*]
Kathmandu University
Dhulikhel, Bagmati, Nepal
prakash@ku.edu.np

ABSTRACT

Word embedding is a technique for understanding the relationship among words by mapping words to numbers. Several kinds of research have been carried out in this field in different languages such as English, Hindi, Bengali etc. but very few works are available in the Nepali language domain. In this work, the word embedding technique using Word2Vec is implemented for Nepali news data. The methodology involved in this work includes Dataset preparation and Word2Vec modelling. Gensim package is used for implementing the Word2Vec model and its output shows the similarity between Nepali words. The work mainly focuses on developing word embedding on Nepali words generated by scraping the health section of Nepali news portals and has shown promising results.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning**;

KEYWORDS

Nepali Word Embedding, Nepali News, Word2Vec

ACM Reference Format:
Bipesh Subedi and Prakash Poudyal. 2022. Word Embedding in Nepali Language using Word2Vec. In *2022 6th International Conference on Natural Language Processing and Information Retrieval (NLPIR 2022), December 16–18, 2022, Bangkok, Thailand.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3582768.3582799

1 INTRODUCTION

With the advancement in deep learning models and the availability of large datasets, various research work has been found in the Natural Language Processing (NLP) domain such as Text Classification, Question Answering, Named Entity Recognition, and Sentiment Analysis [10]. Machine learning algorithms cannot directly understand and process text data. Therefore, the text data must be converted into some kind of numerical representation to perform operations on words. Word embedding is a technique to represent words into numerical forms called vectors [6]. Such representation helps to capture contextual and semantic relationships (such as identifying similarity and dissimilarity, establishing analogies) between words, exhibit visual representation of words etc. [9]. Some of the applications of word embedding includes analyzing survey responses, analyzing verbatim comments, music/video recommendation system etc.

Word embedding in English language is popular and frequently used in natural language processing tasks but only few research work [10] is found in Nepali domain because of less availability of a large text corpus. Moreover, the existing work in Nepali doesn't explain the detailed working of a particular model. In this work, word embedding is performed using Word2Vec model on news data scraped from five news portals (Nagarik, News24, OnlineKhabar, Ratopati, Setopati). Such embeddings shows the similarities between the words that can be used for identifying their features, causes, or effects in the healthcare domain. The main reason behind using Word2Vec is that the words are represented as vectors and their arrangement is done in such a way that the words with similar meanings are grouped together and different words are placed far apart. This work focuses on Word2Vec model and aims to explain its working in detail. The data corpus and model file of this work is available at Kathmandu University Information and Language Processing Research Lab (ILPR)[1] website. The remainder of this paper is arranged in the following order: Section 2 discusses some of the related works, Section 3 shows the methodology used, and Section 4 consists of results and discussions of the work followed by a conclusion and future directions.

2 STATE OF THE ART

Word embeddings are considered the building blocks of many Natural Language Processing (NLP) applications. The word embedding tools , technologies, and pre-trained models are widely available for resource-rich languages such as English and Chinese. Because of the pre-trained models available it is easier for resource-constrained languages such as Portuguese, Arabic, Bengali etc. to perform word embeddings. [10] have presented 25-state-of-the-art word embeddings for Nepali language using models such as GloVe, Word2Vec, fastText, and BERT. But the detailed explanation of the model architectures and its working in the Nepali context are not shown. Similarly, topic modeling in Hindi language using Word2Vec is shown by [14] that clusters semantic space to detect topics from Hindi corpus. [12] has also shown word embedding for Urdu Language using the Word2Vec model. The raw text data collected from various sources such as e-commerce websites, facebook comments, tweets, urdu social media scientists, wikipedia, and relevant online resources were preprocessed and trained using the Gensim package for Skip-Gram Model. Another work done by [7] shows semantic similarities in the English language using Word2Vec. [10] used 279 million word tokens generated from a text corpus which includes news data, wikipedia articles, OSCAR corpus. Likewise, the dataset

[*] Corresponding Author: Prakash Poudyal

[1] https://ilprl.ku.edu.np/downloads/

NLPIR 2022, December 16–18, 2022, Bangkok, Thailand
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-9762-9/22/12.
https://doi.org/10.1145/3582768.3582799

was prepared in [7] using 320,000 English articles from Wikipedia. In contrast, [14] have not explicitly mentioned the source of the dataset but they have mentioned the corpus was created using Hindi sentences. The model was evaluated using intrinsic and extrinsic evaluation metrics in [10]. [14] have not used any metrics but have shown the similarity of the words in the output of the model. [7] have tested the model using WordSim-353, and SimLex-999 whereas [12] concluded that the Word2Vec model performs well for Urdu words compared to other models. On the other hand, [5] have highlighted the use of the Word2Vec model for improving the accuracy of the sentiment analysis task. They have surveyed various literature which uses either the Continuous Bag-of-Words model (CBOW), Skip-Gram, or both. The review shows different models implemented, their evaluation metrics, and approaches followed to perform the task. It is basically a theoretical review and doesnot show any experimental procedure, in contrast to other literature mentioned above.

All these research works have shown word embedding for different languages and have achieved promising results. Other resources such as internet articles and blogs have also shown the process of word embedding using Word2Vec in a more practical way. [6, 9, 11, 13] are some of the articles describing various methods, components, and implementation of Word2Vec model.

Based on these literatures, a Nepali Word Embedding system is proposed in this work which uses Word2Vec for Nepali news data related to health. It aims to provide a practical approach on how word embedding can be achieved in the Nepali context.

3 METHODOLOGY

The methodology opted for this work involves two steps: Dataset Preparation and Model Architecture. The first step is responsible for creating a trainable dataset from scraped text data and the second step is responsible for defining and implementing the Word2Vec model for the prepared data. Figure. 1 depicts the overall system flow for Nepali word embedding system.

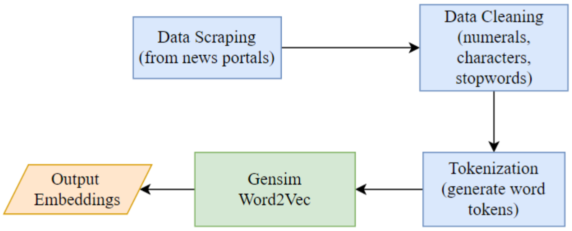


Figure 1: Overall flow of the system

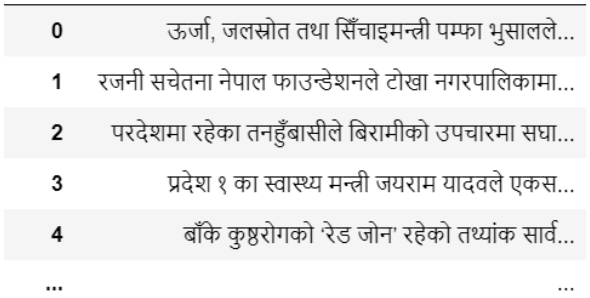
3.1 Dataset Preparation

The dataset preparation task involves two processes: Data Scraping and Data Preprocessing.

3.1.1 Data Scrapping. Data Scraping can be considered as scraping or collecting large volumes of data using web scraping techniques. Web scraping is the process of using bots (web crawlers) to extract data from a website. For web scraping , a python library called 'scrapy' [4] was used to collect data from five news portals (Nagarik[2], News24[3], OnlineKhabar[4], Ratopati[5], Setopati[6]). The scraped raw text data was then saved in a CSV file. Figure 2 shows a sample of the scrapped raw data. In this work, a total of 3,392 articles were scraped from the health section to prepare the text corpus. Table. 1 shows the news portals along with the total number of articles scraped.

Table 1: News portal names with total number of articles retrieved

News portals	Total articles
Nagarik	384
News24	345
OnlineKhabar	2024
Ratopati	14
Setopati	625



```
0        ऊर्जा, जलस्रोत तथा सिँचाइमन्त्री पम्फा भुसालले...
1    रजनी सचेतना नेपाल फाउन्डेशनले टोखा नगरपालिकामा...
2      परदेशमा रहेका तनहुँबासीले बिरामीको उपचारमा सघा...
3        प्रदेश १ का स्वास्थ्य मन्त्री जयराम यादवले एकस...
4          बाँकि कुष्ठरोगको 'रेड जोन' रहेको तथ्यांक सार्व...
...                                                   ...
```

Figure 2: Sample raw corpus data

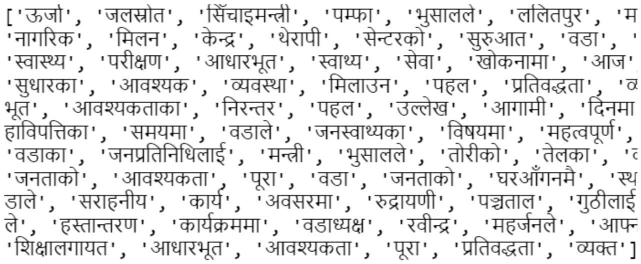
3.1.2 Data Preprocessing. The data preprocessing task involves two major processes: Data Cleaning and Tokenization. In this work, Word2Vec is implemented using the Gensim model [1] which requires training data as 'list of lists' [13]. Therefore, a final 'list of lists ' format dataset is prepared in this step.

Data Cleaning. The data collected after scraping contains many unwanted words,letters, symbols, and characters. Such unwanted data must be removed before feeding the data to the model for better modeling and efficiency. The raw corpus data was cleaned for unwanted text such as Nepali numerals (०,१,२,३,४,५,६,७,८,९), characters such as "।", white spaces, special characters (()#/@;:<>'+ = |!?,'"), and stop words.The stop words were taken from NLTK library.

Tokenization. Tokenizing text is the act of converting strings into tokens and then storing the tokenized content in a single column for simpler processing. This work uses Gensim Word2Vec models

[2] https://nagariknews.nagariknetwork.com/
[3] https://www.news24nepal.tv/
[4] https://www.onlinekhabar.com/
[5] https://www.ratopati.com/
[6] https://www.setopati.com/

which require tokenized words for training. Therefore, tokens are generated in two levels. At first, sentences are tokenized and each word in the sentence is tokenized resulting in a final ‘list of lists’ tokens [12]. It is achieved using “sent_tokenize” and “word_tokenize” methods from the NLTK library [3]. Some of the sample words token after data cleaning is shown in Figure 3.



```
['ऊर्जा', 'जलस्रोत', 'सिँचाइमन्त्री', 'पम्फा', 'भुसालले', 'ललितपुर', 'म
'नागरिक', 'मिलन', 'केन्द्र', 'थेरापी', 'सेन्टरको', 'सुरुआत', 'वडा', '
'स्वास्थ्य', 'परीक्षण', 'आधारभूत', 'स्वाथ्य', 'सेवा', 'खोकनामा', 'आज'
'सुधारका', 'आवश्यक', 'व्यवस्था', 'मिलाउन', 'पहल', 'प्रतिवद्धता', 'व
भूत', 'आवश्यकताका', 'निरन्तर', 'पहल', 'उल्लेख', 'आगामी', 'दिनमा
हाविपत्तिका', 'समयमा', 'वडाले', 'जनस्वाथ्यका', 'विषयमा', 'महत्वपूर्ण',
'वडाका', 'जनप्रतिनिधिलाई', 'मन्त्री', 'भुसालले', 'तोरीको', 'तेलका', 'व
'जनताको', 'आवश्यकता', 'पूरा', 'वडा', 'जनताको', 'घरआँगनमै', 'स्थ
डाले', 'सराहनीय', 'कार्य', 'अवसरमा', 'रुद्रायणी', 'पञ्चताल', 'गुठीलाई
ले', 'हस्तान्तरण', 'कार्यक्रममा', 'वडाध्यक्ष', 'रवीन्द्र', 'महर्जनले', 'आफ्न
'शिक्षालगायत', 'आधारभूत', 'आवश्यकता', 'पूरा', 'प्रतिवद्धता', 'व्यक्त']
```

Figure 3: Sample words token

3.2 Model Architecture

3.2.1 Word2vec Model. The Word2vec algorithm ensures that the words in a similar context have similar embeddings. It starts with a text corpus and outputs word vectors. The system learns vector representations of the words after first creating a vocabulary from the training text input. Each unique word in the sample corpus is allocated to a corresponding vector in the vector space, which can have hundreds of dimensions [2]. Furthermore, terms in the corpus with comparable contexts are clustered in the same space. There are two methods used by Word2Vec to generate word embeddings or vectors: Continuous Bag Of Words (CBOW) and Skip-Gram. Word2vec provides an option to choose between continuous Bag of words and skim-gram. In this work, both CBOW and Skip-Gram were used for word embedding purposes.

Continuous Bag Of Words (CBOW). The CBOW approach uses each of the contexts (neighbours) to predict the current word i.e. to predict the middle word the distributed vector representation of the surroundings words are combined [11, 13]. In CBOW, the word embeddings are attained by solving a fake problem of predicting the current word using its context. The predicted output is then compared with the actual output to update the neural network’s weights. The working is similar for Nepali word embedding. The Continuous Bag Of Words algorithm is trained on the Nepali news text corpus iteratively until embeddings between words are achieved. To illustrate our work, a part of the sentence is taken as an example from the text corpus.

Before preprocessing: “ऊर्जा, जलस्रोत तथा सिँचाइमन्त्री पम्फा भुसालले ... ”

After preprocessing: ['ऊर्जा', 'जलस्रोत', 'सिँचाइमन्त्री', 'पम्फा', 'भुसालले' ...]

To predict ‘सिँचाइमन्त्री’ from the given data, CBOW uses its neighbouring words as shown in Figure. 4. A window size of 5 is used so, two neighbouring words on both sides ('ऊर्जा', 'जलस्रोत', 'पम्फा', 'भुसालले') are taken into consideration. The text data cannot be directly sent to the model so, a one-hot encoding vector corresponding to each word is sent. One-hot encoding is a technique of representing words in a vector form which are feeded to the neural network for better performance. One-hot vector represents each word of the vocabulary uniquely. The CBOW algorithm then trains and predicts the output. If the predicted output is not 'सिँचाइमन्त्री' then the weights are updated based on the error. Once, the training is complete, a weight matrix is obtained which is multiplied by a one-hot encoding vector of the word to obtain the embedding or vector for the word. For instance, the vector embedding for 'सिँचाइमन्त्री' will be, $W \cdot V$ where, W denotes the weight matrix of the neural network and V denotes one-hot encoding of the word 'सिँचाइमन्त्री'. The process is repeatedly performed for every text data in the dataset until embeddings are achieved. Continuous Bag Of Words model is faster to train and has marginal accuracy over Skip-Gram for frequent words [8].

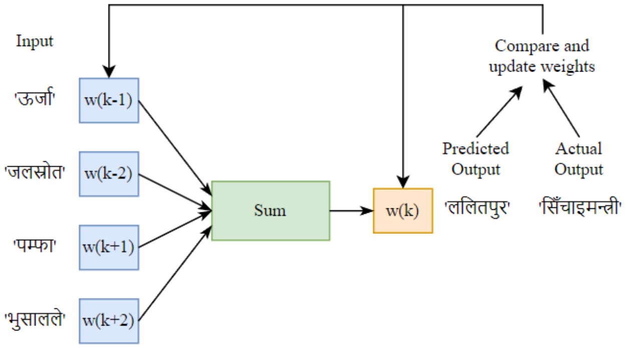


Figure 4: Continuous Bag Of Words (CBOW) architecture

Skip-Gram. The Skip-Gram works as opposed to the Continuous Bag Of Words (CBOW) method. It predicts the context (surrounding words) using the current word, i.e., the input word’s distributed representation is used to predict the context [11, 13]. Similarly, it also solves a fake problem of predicting context using a current word to achieve word embedding as a byproduct. To understand its implementation in our context, consider the previous example. Skip-Gram uses 'सिँचाइमन्त्री' to predict its neighbouring words ('ऊर्जा', 'जलस्रोत', 'पम्फा', 'भुसालले') as shown in Figure. 5. The predicted output is then compared with the actual output to update the weights.

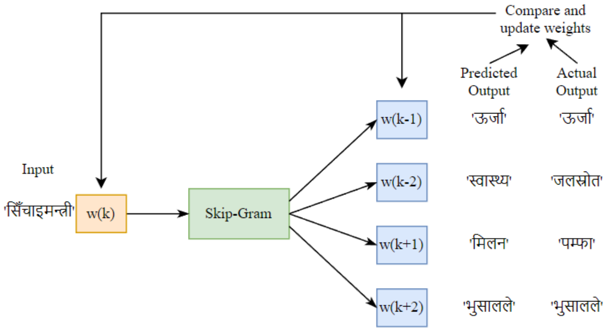


Figure 5: Skip-Gram architecture

Table 2: Model Parameters

Parameters	Value
window	5
min_count	1
workers	4
sg	0/1

The word vectors are obtained by multiplying the weight matrix and one-hot encoding vector of the word as mentioned in the CBOW model. Skip-Gram model is also trained iteratively on the text corpus until final embeddings are achieved. It works well with the small training data and shows the representation of rare words and phrases [8].

3.2.2 Gensim Model Implementation. Gensim package [6] is used to train our CBOW and Skipgram Word2Vec models on the Nepali news data. The model parameters used in this work are shown in Table. 2. A window size of 5 is taken which considers two words on each side of the current/actual word. It is the maximum distance between the actual words and predicted word in a sentence. The min_count denotes the minimum frequency of the words appearing in the sentence. So, in this case a word must be present at-least once in the sentence while training the model. Similarly, many workers threads are used to train the model which denotes the number of partitions while training. To train our model, 4 workers were sufficient. Finally, sg = 0/1 was used for selecting CBOW or Skip-Gram model. All the parameters were kept same for both the models except sg. Before training the model, a word vocabulary of size 74,149 was generated using 'build_vocab(data)' method provided by the Gensim package.

4 RESULTS AND DISCUSSION

The Word2Vec model proposed in this work for Nepali word embedding shows the extent of similarity between words. It also predicts the contextually similar words to a given word. The similarity is shown as a numerical representation between 0 and 1. Continuous Bag Of Words and Skip-Gram have predicted different words for the same input word "अपाङ्गता" as shown in Table. 3 and Table. 4 respectively. The Nepali news data suggests that in most of the cases, people with some kind of disability ("अपाङ्गता") are likely to be financially weak (‘गरिब’), injured (‘घाइते’), sick (‘बिरामी’), alone (‘एकल’), prone to attacks (‘सिकार’), paralyzed (‘विंकलाग’), and so on. They also have a direct relationship with doctors (‘डाक्टरहरु’) and society (‘समाज’) in many senses. Moreover, the results shown by our model also suggest the same. It can be seen that both models perform well in the Nepali context. The Continuous Bag Of Words model has shown a greater degree of similarity to the word “स्पाइनल” (0.98) which is very relevant to “अपाङ्गता”. On the other hand, Skip-Gram has also shown a high degree of similarity to words such as “विंकलाग” (0.93), “डाक्टरहरु” (0.93), “बिरामीहरुको” (0.92) etc. Similarly, the degree of similarity between the words “प्रधानमन्त्री” and “अपाङ्गता” is found to be 0.890 and 0.673 using Continuous Bag Of Words (CBOW) and Skip-ram models respectively.

Table 3: Similar words predicted by CBOW model for the word “अपाङ्गता”

Words	Degree of similarity
'सिकार'	0.98234
'गुमाउनु'	0.98146
'बढिरहेको'	0.98144
'स्पाइनल'	0.98059
'अर्कोतिर'	0.98031
'सयमा'	0.97998
'दलदलमा'	0.97988
'भएकी'	0.97985
'रेफर'	0.97966
'घाइते'	0.97951

Table 4: Similar words predicted by Skip-Gram model for the word “अपाङ्गता”

Words	Degree of similarity
'गरिब'	0.94438
'विंकलाग'	0.93170
'डाक्टरहरु'	0.93062
'समाजले'	0.93052
'बिरामीहरुको'	0.92848
'आय'	0.92527
'विरामीहरु'	0.92470
'भेटिन्छन्'	0.92420
'अझैपनि'	0.92357

From these results, we can conclude that Skip-gram performs well while capturing the contextual details in general as it has predicted more number of relevant contextually similar words than Continuous Bag Of Words (CBOW). In contrast, Continuous Bag Of Words predicts specific relevant words with a high degree of similarity than Skip-Gram. Overall, our work has shown promising results for Nepali news data.

5 CONCLUSION AND FUTURE DIRECTIONS

The Word Embedding for Nepali health news data has been developed by using Gensim Word2Vec word embedding model. The model uses primary datasets created by scraping the health section of five news portals and preprocessing the scraped data. The implementation results in word embedding as a byproduct of solving a fake problem using CBOW and Skip-Gram. The output of this model shows the degree of similarity between the words and is found to have promising results.

This work can be extended further by applying stemming to the words tokens for reducing the words to their root form which improves performance of the system. Furthermore, the data can be increased by scraping more of the nepali news portals as well as considering other domains in addition to health news.

REFERENCES

[1] 2009. *Gensim: topic modelling for human.* https://radimrehurek.com/gensim/models/word2vec.html

[2] 2016. *Word2Vec.* https://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/word2vec.html

[3] 2022. *NLTK :: Natural Language Toolkit.* https://www.nltk.org/

[4] 2022. *Scrapy | A Fast and Powerful Scraping and Web Crawling Framework.* https://scrapy.org/

[5] Samar Al-Saqqa and Arafar Awajan. 2019. The Use of Word2vec Model in Sentiment Analysis: A Survey. In *2019 International Conference on Artificial Intelligence, Robotics and Control (AIRC '19)*, Vol. 157. Association for Computing Machinery, Cairo Egypt, 39–43. https://doi.org/10.1145/3388218.3388229

[6] Zafar Ali. 2019. *A simple Word2vec tutorial.* Retrieved April 18, 2022 from https://medium.com/@zafaralibagh6/a-simple-word2vec-tutorial-61e64e38a6a1

[7] Derry Jatnika Moch Arif Bijaksana and Arie Ardiyanti Suryani. 2019. Word2Vec Model Analysis for Semantic Similarities in English Words. In *4th International Conference on Computer Science and Computational Intelligence 2019 (ICCSCI)*, Vol. 157. Procedia Computer Science, 160–167. https://doi.org/10.1016/j.procs.2019.08.153

[8] Daniel Johnson. 2022. *Word Embedding Tutorial | Word2vec Model Gensim Example.* Retrieved May 18, 2022 from https://www.guru99.com/word-embedding-word2vec.html

[9] Dhruvil Karani. 2018. *Introduction to Word Embedding and Word2Vec.* Retrieved April 18, 2022 from https://towardsdatascience.com/introduction-to-word-embedding-and-word2vec-652d0c2060fa

[10] Pravesh Koirala and Nobal B. Niraula. 2021. NPVec1: Word Embeddings for Nepali - Construction and Evaluation. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*. Association for Computational Linguistics, Bangkok Thailand, 174–184. https://doi.org/10.18653/v1/2021.repl4nlp-1.18

[11] Ria Kulshrestha. 2019. *NLP 101: Word2Vec — Skip-gram and CBOW.* Retrieved May 18, 2022 from https://towardsdatascience.com/nlp-101-word2vec-skip-gram-and-cbow-93512ee24314

[12] Sajadul Hassan Kumhar, Mudasir M. Kirmani, Jitendra Sheetlani, and Mudasir Hassan. 2021. Word Embedding Generation for Urdu Language using Word2vec model. *Materials Today: Proceedings* (2021). https://doi.org/10.1016/j.matpr.2020.11.766

[13] Zhi Li. 2019. *A Beginner's Guide to Word Embedding with Gensim Word2Vec Model.* Retrieved April 18, 2022 from https://towardsdatascience.com/a-beginners-guide-to-word-embedding-with-gensim-word2vec-model-5970fa56cc92

[14] Sabitra Sankalp Panigrahi Narayan Panigrahi and Biswajit Paul. 2018. Modelling of Topic from Hindi Corpus using Word2Vec. In *2018 Second International Conference on Advances in Computing, Control and Communication Technology (IAC3T)*. IEEE, 97–100. https://doi.org/10.1109/IAC3T.2018.8674031