www.revistageintec.net
ISSN: 2237-0722
revista GEINTEC Gestão, Inovação e Tecnologias

Evaluating Word Embedding Models for Malayalam

Merin Cherian[1]; Kannan Balakrishnan[2]
[1]Federal Institute of Science and Technology, Angamaly, India.
[1]merincherian123@gmail.com
[2]Cochin University of Science and Technology, Kochi, India.
[2]mullayilkannan@gmail.com

Abstract
An evaluation of static word embedding models for Malayalam is conducted in this paper. In this work, we have created a well-documented and pre-processed corpus for Malayalam. Word vectors were created for this corpus using three different word embedding models and they were evaluated using intrinsic evaluators. Quality of word representation is tested using word analogy, word similarity and concept categorization. The testing is independent of the downstream language processing tasks. Experimental results on Malayalam word representations of GloVe, FastText and Word2Vec are reported in this work. It is shown that higher-dimensional word representation and larger window size gave better results on intrinsic evaluators.

Key-words: Malayalam, Word Embedding, Intrinsic Evaluation.

1. Introduction

Malayalam is a Dravidian Language spoken in Kerala. Natural language processing in Malayalam is still in its infancy. The unavailability of linguistic resources for Malayalam is the primary hindrance to research. Publicly available corpora are available for languages like English. However, for languages like Malayalam, a large corpus that is free is not yet available. The vast majority of the people in Kerala learn English as a second language. As a result, they are can use the web and its resources using English. This has been one of the reasons for the low availability of web resources in Malayalam. In addition to the low availability of resources, Malayalam is inflectional and agglutinative (1). This makes basic NLP tasks like tokenization, stemming, morphological analysis, etc. very difficult. The saturation level of the language is also higher (1), and as a result, a larger text corpus is

required to provide a good span of the words in the language. Training a language model for Malayalam even on a smaller corpus is also computationally expensive due to the number of words and characters that make up the language.

Word embedding is a popular tool that enables word representations. Words or phrases are represented as real value vectors that are learned from a large corpus of unlabelled text (2). It can illustrate syntactic and semantic relations between words. Word tokens are encoded into a vector in an N-dimensional space, where each dimension encodes some semantics of the language. Word embedding helps in numerical modelling of the semantic importance of a word. This permits mathematical operations to be performed on it. Currently, word embeddings are available in two types- (i) Static Embedding and (ii) Dynamic/Contextualized Embedding. Static embeddings are static in the sense that they do not change with respect to the context once the training is done. The most popular static embedding techniques are GloVe, FastText, and Word2vec. The problem with static embedding is that a polysemous word has the same representation in every context. To overcome this problem many methods were proposed to learn embedding from the context. Pre-trained language models like BERT, ELMo, and GPT-2 can be used to extract contextualized word embedding. Word embeddings are used effectively in a wide range of NLP tasks like machine translation, POS tagging, Named Entity Recognition, syntactic parsing and semantic labelling. Word embedding can be evaluated using either intrinsic methods or extrinsic methods (3). Intrinsic evaluators assess the syntactic and semantic relationship between words directly. Extrinsic evaluators utilize the word embedding vectors as an input to the NLP tasks.

In this paper, our objective is to create a corpus for Malayalam and generate static word embedding models and evaluate those using intrinsic evaluators. Section 2 describes the process of creating the corpus. A brief review of the static embedding models discussed in this paper is covered in Section 3. Intrinsic evaluators are discussed in Section 4. Experimental results on intrinsic evaluators are described in Section 5. Future work is presented in Section 6.

2. Corpus Creation

A corpus is outlined as a database of statistically sampled language (4) created for performing linguistic investigation, description, application, and analysis. A well-documented corpus is an essential linguistic resource because of the huge size and information, diverse content, broad spectrum of representation, usability and verifiability. Creating a corpus is a slow and tedious process, especially because of the unavailability of linguistic tools like spell checker, tokenizer which cover the entirety of

the Malayalam language. Already existing Malayalam corpora include Malayalam Monolingual Text Corpus with 31,000 sentences of the general domain, Malayalam Monolingual Health text Corpus with 23000 sentences and Malayalam Monolingual Tourism Text corpus with 12500 sentences from TDIL[1]. A Gold Standard Malayalam Raw Text Corpus (3; 4) with 63, 70,954 words and 1,119 titles were developed under Linguistic Data Consortium for Indian Languages[2]. But this corpus is not available for free.

The majority of the Malayalam texts available on the web are either on news websites or blogs which are copyrighted. We crawled news websites like manoramaonline[3], kairalinews[4], asianetnews[5], oneindia[6] to obtain a major portion of our corpus. Malayalam text extracted from school text books[7] for classes 1 to 12, Malayalam Bible and Malayalam books with Creative Commons license obtained from internet archive[8] are also part of the corpus. Our corpus also includes texts from WikiPedia, WikiGrandhasala which are available in the public domain. Books obtained from internet archives include essays, novels, children's stories, etc. A few editions of Manorama yearbook, Bashaposhini and Manorama weekly were also obtained from the publisher for this purpose. Optical character recognition was done on these books using Tesseract and further processing was done to remove unwanted characters and text.

Our Malayalam corpus has 82 unique characters, 30 million tokens, 3.8 million unique words. It accounts for about 1 GB of text data which is stored as text files using Unicode encoding. The corpus is also indexed with metadata for easier retrieval and verifiability.

2.1. Corpus Preprocessing

The text corpus is encoded using UTF-8 encoding and stored in .txt format. Unwanted characters, emoticons, characters in other languages were removed using regular expression processing. Words ending with ൺഒഠ were replaced with ൻഠ, symbols like ഭാ൦ were replaced with ഭ൦. All punctuations except the period symbol were removed. Blank lines, leading and trailing spaces, multiple spaces, carriage returns were deleted. Text collected from every source is well documented and also

[1] http://tdil-dc.in/
[2] https://data.ldcil.org/
[3] https://www.manoramaonline.com/
[4] https://www.kairalinewsonline.com/
[5] https://www.asianetnews.com/
[6] https://malayalam.oneindia.com/
[7] https://samagra.kite.kerala.gov.in/
[8] https://archive.org/

kept in separate files for easy retrieval as well as for verification purposes. The metadata for our corpus includes file name, category of text, the title of the text, numbers of words and letters in the text file. Poetic texts were removed from the corpus as well as typographical errors and spelling mistakes were corrected. Sentence tokenization was done on the noise removed text before creating word embedding models for Malayalam.

2.1.1. The Bible, School textbooks, Manorama Year Book, Bashaposhini and Manorama Weekly

Optical character recognition was done on these books using the Tesseract tool. A new line was added after every sentence. Chapter headings and numbers were removed. Sentence tokenization as well as chapter heading and number removal was done. Spelling mistakes were corrected manually as well as using the aspell spell checker tool. Lines with illegible characters were deleted. Corrections were made on words that were broken because they appeared in the next line. The abbreviations broken as a new sentence as a result of sentence tokenization were also corrected. Verse numbers were removed from the Bible. Texts which were part of advertisements were removed from the magazines and weeklies. The Bible alone accounted for 10MB of text with 31000 lines. School Textbooks contributed 70000 lines of text amounting to 15MB of data. Manorama Year Book, Bashaposhini and Manorama Weekly together formed 1 lakh lines of text creating 25MB of text.

2.1.2. Wikipedia Dump, Wiki Grandhasala, News Websites

The Wikipedia dump was obtained using a wiki extractor while Wiki Grandhasala and news websites were crawled. Links and references were removed from the crawled text and sentence tokenization was done. Abbreviations were also corrected. Emoticons, English, Chinese, Arabic and punctuation marks other than period symbols were removed. Wikipedia dump, Wiki Grandhasala, News websites and some blogs and Malayalam websites make up 950 MB of data. According to (4) our corpus is a special corpus, which satisfies the features like quantity, quality, simplicity, verifiability, augmentation, documentation and management. Our corpus is the only corpus presently available in Malayalam with a size of 30 million tokens, 3.8 million unique words. The corpus is authentic in that it was collected from the public domain. Other texts were collected after communicating with the publishers and getting approval. This is a simple raw corpus text stored in Unicode format. The corpus can be verified and scrutinized so that the users can certify that the corpus is a true reflection of Malayalam. The corpus can also be augmented so that it can grow with time. Our corpus is

comprehensively documented so that the sources of texts are preserved. The corpus is also properly managed so that the text files are arranged to help in easier information retrieval and maintenance.

3. Word Embedding Models

Word embedding models are used to obtain syntactic and semantic meanings from an unlabeled corpus. Word embedding representations have evolved in the past few years. These algorithms can train models based on character level, word level, phrase level, sentence level, and paragraph-level or document level information. Word embedding models can be static or dynamic. A static word embedding maps each word type to a dense single vector of lower dimension than the vocabulary size. This mapping cannot handle polysemy. The desirable properties of word embedding models include non-conflation, robustness against lexical ambiguity, and demonstration of multifacetedness (5), reliability (6), and good geometry (7). In this section, we give a review of the state-of-the-art static word embeddings.

3.1. Word2Vec

Two architectures namely Continuous Bag of Words (CBOW) and Skip-gram model were put forward by Mikolov et.al (8) to reduce the computational complexity of distributed representations like NNLM(9), SENNA(10). CBOW and Skip-gram models can be implemented using Word2Vec. It can either predict the words which appear in the context of a target word or given the surrounding context it can predict a target word. Word2Vec creates a classifier which is a dense vector representation of the words. Words in similar contexts will have similar representations.

The CBOW model uses context words as input and outputs the target word. The input is encoded as a one-hot vector and output is predicted using softmax. The skip-gram model takes the target word as input and predicts the surrounding context words. The target word is encoded as a one-hot vector and output is predicted using softmax. The projection layer between the input and output layer of CBOW and Skip-gram model averages the vectors obtained by mapping each one-hot vector to a dense D-dimensional vector.

Thus the CBOW model can be expressed as

$$P(w_i|w_{i-c}, w_{i-c+1}, \ldots, w_{i-1}, w_{i+1}, \ldots, w_{i+c-1}, w_{i+c}), \quad (1)$$

where w_i is a word at position i and c is the window size.

The skip-gram model is expressed as

$$P(w_{i-c}, w_{i-c+1}, \ldots, w_{i-1}, w_{i+1}, \ldots, w_{i+c-1}, w_{i+c}|w_i) \quad (2)$$

3.2. GloVe

The global vector model uses a word co-occurrence matrix to obtain word representation (11). The word co-occurrence statistics of a corpus form the basis for unsupervised learning in word representations. GloVe uses the global corpus statistics for creating the word vectors and to thus derive the semantic relationships between words. The co-occurrence matrix is generated by calculating the frequency of a word co-occurring with another in the training corpus. The statistics are collected by passing through the entire corpus once. The glove works under the intuition that the ratios of word-word co-occurrence probabilities can encode meaning.

Given a corpus of vocabulary size N and its co-occurrence matrix X where Xij denotes the number of times word j occurs in the context of the word i

$$P_{ij} = P(j|i) = X_{ij}X_i \quad (3)$$

is the probability that the word j appears in the context of the word i. The loss function in GloVe is defined as follows

$$L = \Sigma_{i,j=1}^{|V|} g(X_{ij})\left(C(w_i)^T C(w_j) + b_i + b_j - logX_{ij}\right)^2 \quad (4)$$

where $|V|$ is the vocabulary size, and $g(x)$ is a weighting function that manages the imbalance caused by rare and frequent words.

3.3. FastText

To address the morphology of words in vector representation FastText (12; 13) was introduced. Here each word is represented as the average of the vectors associated with each of the character n-grams of a word. A bag of character n-grams forms the sub-word units of a word. FastText uses the continuous Skip-gram model. It improves the embedding of rarely used words as well as the performance on syntactic tasks. FastText can train models using negative sampling, softmax, or hierarchical softmax loss functions.

4. Intrinsic Evaluators

Evaluators compare the word embedding models using quantitative and representative measures. Since it is difficult to find a standard way to evaluate the abstract characteristics of a word

embedding model, a word embedding evaluator may possess the properties like good testing data, comprehensiveness, high correlation, efficiency, and statistical significance (2) for obtaining better metrics.

Word representations can be evaluated using the syntactic or semantic relationship between words directly. These intrinsic evaluations do not depend on any NLP downstream tasks. The scores for intrinsic evaluation are obtained by testing the word vectors against a selected set of query terms and the semantically related target words. Intrinsic evaluation can be absolute evaluation or comparative evaluation (1; 2). In absolute evaluation, the benchmark relationship scores between query and target words are collected in advance and tested with word vectors. In comparative intrinsic evaluation, human evaluators assess the quality of word representations. In this section, the absolute intrinsic evaluations used for evaluating Malayalam word embedding models are discussed.

4.1. Word Similarity

Word semantic similarity compares the distance between word vector representations and semantic similarity judged by humans. Similarity measurement scores how well the word embeddings capture the word similarities as perceived by the humans. The most commonly used evaluator is cosine similarity.

$$cos(w_x, w_y) = \frac{w_x . w_y}{||w_x|| \, |w_y|)} \quad (5)$$

where the two word vectors are given by w_x and w_y and $||w_x||$, $||w_y||$ form the $\ell 2$ norm.

Commonly used word similarity datasets include WS-353 (14), WS-353-SIM (15), WS-353-REL (15), MC-30 (16), RG-65 (17), Rare-Word (RW) (18), MEN (19), MTurk-287 (20), MTurk-771 (21), YP-130 (22), SimLex-999 (23), Verb-143 (24), SimVerb-3500 (25). The number of word pairs in each dataset is given along with the dataset name. The MEN data set has 3000 word pairs. The most popular similarity dataset are WS-353 and RareWord. Word similarity evaluator computationally inexpensive and can be used to test a word embedding model for distinguishing lexical ambiguity (2).

4.2. Word Analogy

In word analogy evaluation given three words w,x and y, the word vectors can predict the word z, such that the relationship between w and x is the same as that of y and z. It can be represented as $w : x : : y : z$. For example

$$Man : Woman : : King : Queen$$

Received: 17.06.2021 – Accepted: 18.07.2021

Lexical semantic relations like synonyms and antonyms cannot be described using word analogy evaluations (2). This evaluation is more accurate when the distance between the three source vectors is close to the target vector. Subjectivity is another problem with this evaluation. Since word embeddings do not encode human reasoning the word vectors may find a different relationship than that of humans. Google analogy dataset (8) and MSR dataset (26) are the commonly used dataset in analogy evaluation.

4.3. Concept Categorization

Word vector embeddings are evaluated using clustering in concept categorization. The given set of words are grouped into different classes using clustering. The AP dataset (27), the BLESS dataset (28) and the BM dataset(29) are the popular datasets used in concept categorization. The AP dataset has 402 words in 21 different categories, the BM has 5321 words grouped into 56 categories, BLESS dataset divides 200 words into 27 classes. The ESSLLI dataset consists of 44 concrete nouns divided into 6 semantic categories.

5. Results and Discussion

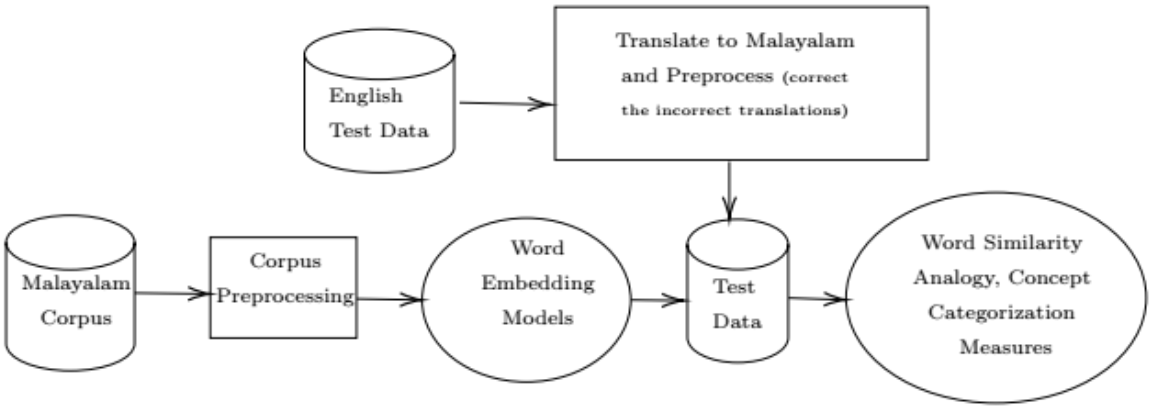
We evaluate three Malayalam word embedding models using intrinsic evaluators like word similarity, word analogy and concept categorization. The word embedding models generated are Word2Vec (CGNS and CBOW), GloVe and FastText (CGNS and CBOW). Our Malayalam dataset is pre-processed and consists of 30 million tokens. The official toolkits Word2Vec[9], Glove[10] and FastText[11] were used for training. For each of the embedding models, word vectors of 100 dimensions and 300 dimensions were generated. The window size for each dimension was set at 15 and 5. The vocabulary to create the models was generated at two frequencies, 1) if the word occurred at least 5 times, 2) if the word frequency is one. The vocabulary size was 3841593 when all the words were considered and the vocabulary size when the minimum word occurrence frequency was five is 444788. The models generated are given in Table 1. The training was done on Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz with 32 GB memory. Word embeddings for 300 dimensions and vocabulary size one could not be generated due to memory constraints. A flow diagram of our evaluation system is shown in Figure 1.

[9] https://code.google.com/archive/p/word2vec/
[10] https://nlp.stanford.edu/projects/glove/training
[11] https://github.com/facebookresearch/fastText

Table 1- List of the different Malayalam Embedding Models Generated

Models	Dimension	Window Size
Word2Vec, Glove, FastText	100	15
		5
	300	15
		5

Figure 1 - Workflow for Evaluating Malayalam Word Embeddings



The Malayalam language does not have a standard dataset to perform an sintrinsic evaluation on our embedding models. So we created an evaluation dataset from the existing benchmark datasets. The standard evaluation datasets were translated to Malayalam. Initial translation was done with the help of Google Translate. In the next stage, each dataset was manually processed to correct the incorrect translations from the first step. A few of the incorrect and their correct translations are given in Table 2.

Table 2- Incorrect Translation

English Word	Incorrect Malayalam Translation	Correct Malayalam Translation
Protégé	പ്രോട്ടീജ്	ശിഷ്യൻ
client	ക്ലയൻറ്	ഇടപാടുകാരൻ
promiscuity	പ്രോമിക്യൂറ്റി	സങ്കീർണ്ണത
shuffles	ഷഫിളുകൾ	ഇളക്കുന്നു
generate	ജനറേറ്റുചെയ്യുക	ഉണ്ടാക്കുക

Some of the translations are incorrect due to the context in which they appear. For example the word pair dark: light was translated to ഇരുട്ട്: ഭാരം കുറച്ചു. The second word means light in terms of weight in Malayalam. A few of such incorrect translations and their corrections are given in Table 3.

Received: 17.06.2021 – Accepted: 18.07.2021

Table 1- Incorrect Translation on Context

English Word Pair	Incorrect Malayalam Translation	Correct Malayalam Translation
play: plays	കളിക്കുക: നാടകങ്ങൾ	കളിക്കുക: കളിക്കുന്നു
live: lived	തത്സമയം: ജീവിച്ചു	ജീവിക്കുക: ജീവിച്ചു
falling: land	വീഴുക: ഭൂമി	വീഴുക: പതിക്കുക
tip: waiter	അഗ്രം: വെയിറ്റർ	ലഘുപാരിതോഷികം:പരിചാരകൻ
work: works	ജോലി: പ്രവർത്തിക്കുന്നു	പ്രവർത്തിക്കുക: പ്രവർത്തിക്കുന്നു

5.1. Experimental Setup

5.1.1. Word Similarity

The datasets used for word similarity evaluation of our Malayalam word embedding models are WS-353 (14), WS-353-SIM (15), WS-353-REL (15), RG-65 (17), Rare-Word (RW) (18), MEN (19), MTurk-287 (20), and SimLex-999 (23). All the datasets were translated as per the two-stage process mentioned above.

5.1.2. Word Analogy

The word analogy evaluation is done using the Google analogy dataset (8), MSR dataset (26) and SemEval-2012 (30). SemEval-2012 task dataset consists of 79 categories of graded relational similarity ratings (30). The task involved comparing two pairs of words A and B which are similar to C and D and the degrees of relational similarity between the reference word pair and the other pairs have to be found. We have scored the dataset using platinum scoring labels released by the task organizers. We removed the word pairs under the superlative category from the Google analogy dataset because Malayalam does not have an inflectional - est marker instead, a periphrastic marker ഏറ്റവും which means ‘most’ is used (31). The method used for analogy evaluation was $3CosAdd$ which uses cosine similarity to normalize vector lengths.

5.1.3. Concept Categorization

Four datasets namely the AP dataset (27), the BLESS dataset (28), the BM dataset (29) and the ESSLLI-2008 (32) were used for concept categorization evaluation. The dataset was directly translated to Malayalam using Google translate because the dataset contained only simple words without any inflections. The ESSLLI-2c dataset consists of 45 verbs belonging to 9 semantic classes, ESSLLI-2b

consists of 40 nouns extracted from the MRC Psycholinguistic Database and ESSLLI-1a consists of 44 concrete nouns, belonging to 6 semantic categories.

5.2. Experimental Results and Discussion

The cosine similarity measure obtained for a few word pairs in Malayalam for the different models at 300 dimensions as well as word analogy results for a few questions are given in Table 4. A comparative performance of word similarity, word analogy and concept categorization evaluation scores are given in Table 5. In Table 5 the model, $W_{100}(S)^{(w5,v1)}$ denotes the embedding model Word2Vec of 100 dimensions, Skip Gram Model, generated using window size 5 and vocabulary word minimum frequency one. Similarly, $F_{100}(C)^{(w5,v1)}$ stands for FastText embedding of 100 dimensions, CBOW model, with window size 5 and vocabulary word minimum frequency one and $G_{100}^{(w5,v1)}$ denotes GloVe model with the specified parameters. We can see from the results that similarity evaluation yielded better results for Skip-gram models. Higher dimension vectors showed performance improvement. Word2Vec performed better on lower vocabulary size and larger window size and FastText on smaller window size. The performance of Glove was bad compared to the other two embedding models. Generally, FastText gave superior results on all the datasets in word similarity evaluation.

Table 4- Cosine Similarity Measures and Word Analogy Results of various Models

Model	Word Similarity - Word Pair	Cosine Similarity	Word Analogy - Question	Answer
Word2Vec SkipGram Model	'ഇന്ത്യ ' and 'രാജ്യം'	0.726	'പുരുഷൻ', 'രാജാവ്':'സ്ത്രീ' ,?	ബ്രിട്ടീഷുകാർ
	'പുരുഷൻ ' and 'സ്ത്രീ'	0.584	'പാരിസ്', 'ഫ്രാൻസ്': 'ലണ്ടൻ' , ?	ജപ്പാൻ
	'രാത്രി ' and 'ഇരുട്ട്'	0.446	'പുരുഷൻ', 'പുരുഷന്മാർ': 'സ്ത്രീ', ?	സ്ത്രീകൾ
	'ഇന്ത്യ ' and 'കാർ'	0.270	'നടക്കുക', 'നടക്കുന്നു': സംസാരിക്കുക',?	മറ്റും
Word2Vec CBOW Model	'ഇന്ത്യ ' and 'രാജ്യം'	0.737	'പുരുഷൻ', 'രാജാവ്':'സ്ത്രീ' ,?	ബ്രിട്ടീഷുകാർ
	'പുരുഷൻ ' and 'സ്ത്രീ'	0.575	'പാരിസ്', 'ഫ്രാൻസ്': 'ലണ്ടൻ' , ?	കാനഡ
	'രാത്രി ' and 'ഇരുട്ട്'	0.391	'പുരുഷൻ', 'പുരുഷന്മാർ': 'സ്ത്രീ', ?	സ്ത്രീകൾ
	'ഇന്ത്യ ' and 'കാർ'	0.224	'നടക്കുക', 'നടക്കുന്നു': സംസാരിക്കുക',?	ഉയിർത്ത
FastText SkipGram Model	'ഇന്ത്യ ' and 'രാജ്യം'	0.498	'പുരുഷൻ', 'രാജാവ്':'സ്ത്രീ' ,?	രാജാവെ
	'പുരുഷൻ ' and 'സ്ത്രീ'	0.526	'പാരിസ്', 'ഫ്രാൻസ്': 'ലണ്ടൻ' , ?	ബ്രിട്ടൻ
	'രാത്രി ' and 'ഇരുട്ട്'	0.446	'പുരുഷൻ', 'പുരുഷന്മാർ': 'സ്ത്രീ', ?	സ്ത്രീകൾ
	'ഇന്ത്യ ' and 'കാർ'	0.222	'നടക്കുക', 'നടക്കുന്നു': സംസാരിക്കുക',?	സംസാരിക്കുന്നു
FastText CBOW Model	'ഇന്ത്യ ' and 'രാജ്യം'	0.454	'പുരുഷൻ', 'രാജാവ്':'സ്ത്രീ' ,?	രാജാവോ
	'പുരുഷൻ ' and 'സ്ത്രീ'	0.502	'പാരിസ്', 'ഫ്രാൻസ്': 'ലണ്ടൻ' , ?	ലണ്ടൻ
	'രാത്രി ' and 'ഇരുട്ട്'	0.285	'പുരുഷൻ', 'പുരുഷന്മാർ': 'സ്ത്രീ', ?	സ്ത്രീകൾ
	'ഇന്ത്യ ' and 'കാർ'	0.142	'നടക്കുക', 'നടക്കുന്നു': സംസാരിക്കുക',?	സംസാരിക്കുന്നു
GloVe	'ഇന്ത്യ ' and 'രാജ്യം'	0.489	'പുരുഷൻ', 'രാജാവ്':'സ്ത്രീ' ,?	രാജാവിന്റെ
	'പുരുഷൻ ' and 'സ്ത്രീ'	0.483	'പാരിസ്', 'ഫ്രാൻസ്': 'ലണ്ടൻ' , ?	ജർമനി
	'രാത്രി ' and 'ഇരുട്ട്'	0.205	'പുരുഷൻ', 'പുരുഷന്മാർ': 'സ്ത്രീ', ?	സ്ത്രീകൾ
	'ഇന്ത്യ ' and 'കാർ'	0.262	'നടക്കുക', 'നടക്കുന്നു': സംസാരിക്കുക',?	മുടന്തർ

Received: 17.06.2021 – Accepted: 18.07.2021

From Table 5, it can be seen that all the models gave poor results for analogy evaluation. The FastText model gave the best performance when the window size was small and the vocabulary size was large. The SemEval-2012 dataset gave better results compared to the other two datasets.

Table 5- Comparison of Word Similarity Score x 100, Word Analogy Score x 100, Concept Categorization Score x 100 of different Datasets

	Word Similarity								Word Analogy			Concept Categorization					
Model	MEN	WS-353	WS-353-REL	WS-353-SIM	SimLex-999	RW	RG-65	MTurk-287	Google	MSR	SemEval2012	AP	BLESS	BM	ESSLI_2c	ESSLI_2b	ESSLI_1a
$W_{100}^{(w5,v1)}(S)$	45.5	41.0	33.8	42.5	25.2	25.9	52.8	48.5	3.6	5.5	10	51.2	47	25.4	51.1	72.5	72.7
$W_{100}^{(w5,v1)}(C)$	40.1	40.2	33.9	44.2	20.8	25.1	39	50.4	3	4.5	10.9	50.2	44	23.3	46.7	75	70.5
$G_{100}^{(w5,v1)}$	28	26.6	22.2	26.9	19.7	15.1	37	25	0	0.1	9.5	38.8	42.5	19.1	40	70	72.7
$F_{100}^{(w5,v1)}(S)$	49.7	47.1	41.8	49.8	26.9	27.9	54.7	54.8	6.3	7.9	15	46.5	43	23.7	46.7	72.5	63.6
$F_{100}^{(w5,v1)}(C)$	32.1	34.6	30.5	40.5	22.4	22.8	42.7	32.2	3.6	7	10.9	41	29.5	17.1	35.6	75	54.5
$W_{100}^{(w15,v1)}(S)$	49.4	44.5	37.7	48	25.5	23.6	57.1	50.9	3.9	5	10.3	49	54	25.8	44.4	72.5	70.5
$W_{100}^{(w15,v1)}(C)$	41.5	43	38.5	46.1	20.6	22.4	37.1	53.9	2.9	4	11.7	48.3	45.5	24.7	48.9	70	72.7
$G_{100}^{(w15,v1)}$	31.6	26.9	24	28	21.8	15	28.7	32.1	0.3	0.8	9	39.8	48.5	20.5	48.9	75	70.5
$F_{100}^{(w15,v1)}(S)$	48.7	45.8	41.7	47.6	25.4	25.2	56.8	56	4.7	5.8	11.3	48	42.5	23.7	46.7	77.5	65.9
$F_{100}^{(w15,v1)}(C)$	34.4	39.5	38.2	43	21.6	23.2	41.3	34.2	3.8	7	11.6	40	32.5	18.2	37.8	70	52.3
$W_{100}^{(w5,v5)}(S)$	48.2	44.6	38.8	46.7	26.1	22.6	54	47.6	4	5.9	10.2	50.7	50.5	25.3	48.9	72.5	72.7
$W_{100}^{(w5,v5)}(C)$	43.7	39.7	33.5	44.2	22.3	25.3	35	49.6	3.7	5.2	12	51	47.5	23.9	53.3	75	75
$G_{100}^{(w5,v5)}$	27.4	28.9	23.4	32.6	18.7	18.3	31.9	30.8	0.9	2	10.8	41.5	48.5	18.1	40	72.5	70.5
$F_{100}^{(w5,v5)}(S)$	49.5	45.7	42.3	48.2	25.8	24.4	52.1	50.6	8.6	10.6	9.1	50.5	49.5	23.8	44.4	70	75
$F_{100}^{(w5,v5)}(C)$	34.9	37.6	36.3	41.1	22.4	21.7	39	33.9	6.7	11.2	10	43	34	17.6	42.2	72.5	61.4
$W_{300}^{(w5,v5)}(S)$	42.6	44.5	36.9	46.5	27.3	20.4	58.3	42.2	2.6	5.8	6.3	46.3	50	24.6	51.1	70	70.5
$W_{300}^{(w5,v5)}(C)$	43.5	41.8	36.7	46.1	22.8	25.5	37.8	50	4	6.1	10.4	51.5	46.5	23.3	48.9	70	72.7
$G_{300}^{(w5,v5)}$	27.9	30.7	24.7	35.6	18.4	19.1	35.8	31.4	0.9	1.5	10.1	41.3	52	18.1	44.4	67.5	65.9
$F_{300}^{(w5,v5)}(S)$	47.5	47.4	40.1	50.7	27.9	24.4	53.6	46.8	10.5	12.3	11.2	49	45.5	22.5	44.4	72.5	63.6
$F_{300}^{(w5,v5)}(C)$	35.2	38.7	34.9	43	23	22.2	42.5	30.8	8.6	13.7	10.7	37.3	31.5	16.7	40	75	59.1
$W_{100}^{(w15,v5)}(S)$	49.8	43.6	37.9	48.1	26.5	18.5	51.3	51.3	4.2	5.4	8.3	48.8	51	24.2	48.9	72.5	72.7
$W_{100}^{(w15,v5)}(C)$	44	41	36.1	43.8	21.1	23	34.6	50.8	3.2	4.8	10.8	49.8	47	23.4	46.7	70	75
$G_{100}^{(w15,v5)}$	32.9	29.2	24.6	33.5	18.9	20.1	31.8	38.1	1.8	3.5	9.4	46.3	49	20.5	46.7	70	77.3
$F_{100}^{(w15,v5)}(S)$	46.8	42.5	41.3	43.4	23.1	21	49.7	51.1	6.5	7.8	8.2	48.8	47.5	23.8	44.4	72.5	70.5
$F_{100}^{(w15,v5)}(C)$	35.7	42	41.1	44.7	22.2	20.8	41.6	33.6	6.4	11.1	8.9	38.3	31.5	17.6	42.2	67.5	63.6
$W_{300}^{(w15,v5)}(S)$	32.3	35.7	30.2	36.3	21.6	26.4	49.2	31.6	3.2	5	10.4	45.3	42.5	22.3	42.2	72.5	70.5
$W_{300}^{(w15,v5)}(C)$	27.3	35.8	28.2	40.9	17.7	22.1	42.3	37.3	1.8	2.7	10.7	44	39.5	21.5	46.7	77.5	70.5
$G_{300}^{(w15,v5)}$	34.5	30.3	23	35.8	19.8	19	38.9	36	2.1	4	7.9	43.8	52.5	20.1	48.9	70	75
$F_{300}^{(w15,v5)}(S)$	47.7	46.2	41.6	48.5	25.7	22.2	57.1	46.6	10	11.3	9.6	46.8	48	22.9	42.2	70	65.9
$F_{300}^{(w15,v5)}(C)$	37.2	42.5	39.5	46.1	22.8	21.8	42.1	32.7	8.4	13.5	9.7	41.3	32	17.1	35.6	72.5	54.5

Malayalam doesn't have a large corpus. A larger training corpus will lead to better embedding models and thereby provide good results. The unavailability of open digital documents for Malayalam makes the construction of a larger corpus difficult. The embedding dimension of 300 gives a better representation of the language. Another problem is the absence of an evaluation dataset. The results might have been better if there was a benchmark dataset created by language experts.

6. Conclusion and Future Work

In this paper, we provided an evaluation of the static embedding models for the Malayalam corpus. The embeddings were created on a corpus of size 1 GB. The corpus was created from Malayalam text obtained from the internet, literature books, magazines and school textbooks. The corpus was preprocessed and metadata for the corpus was stored for easier retrieval. Word embeddings were generated for three embedding modes for dimensions of 100 and 300 as well as for different window and vocabulary sizes. We have found that FastText embeddings gave far better results than other word embeddings. These embeddings can be used for further downstream tasks like NER, text classification (1), etc.

Word embedding quality is affected by factors like the size of the corpus, evaluation dataset etc. We expect to create a Malayalam corpus of larger size and create pre-trained vectors which can be efficiently used for downstream tasks. Benchmark datasets in Malayalam for various tasks like similarity, analogy, concept categorization as well as for downstream tasks like named entity recognition, classification etc. should also be created. Dynamic word representations have achieved the state-of-the art results in several NLP tasks in English. Further work on generating and evaluating dynamic word embeddings for Malayalam can advance the language processing tasks for Malayalam in the right direction.

References

Ramamoorthy, L., Narayan Choudhary, Saritha S.L., Rejitha K.S., Sajila S., 2019. *A Gold Standard Malayalam Raw Text Corpus.* Central Institute of Indian Languages, Mysore.

Wang, Y., Hou, Y., Che, W., Liu, T., 2020. From static to dynamic word representations: a survey. *Int. J. Mach. Learn. Cybern. 11,* 1611–1630. https://doi.org/10.1007/s13042-020-01069-8.

Wang, B., Wang, A., Chen, F., Wang, Y., Kuo, C.C.J., 2019. Evaluating word embedding models: Methods and experimental results. *APSIPA Trans. Signal Inf. Process.* 8, 1-14. https://doi.org/10.1017/ATSIP.2019.12.

Dash, N.S., Arulmozi, S., 2018. *History, features, and typology of language corpora, History, Features, and Typology of Language Corpora.* https://doi.org/10.1007/978-981-10-7458-5.

Yaghoobzadeh, Y., Schütze, H., 2016. Intrinsic subspace evaluation of word embedding representations, in: *54th Annual Meeting of the Association for Computational Linguistics, ACL* 2016 - Long Papers. https://doi.org/10.18653/v1/p16-1023.

Hellrich, J., Hahn, U., 2017. Don't get fooled by word embeddings: better watch their neighborhood. Digit. Humanit. 2017— *Conference Abstr. 2017 Conf. Alliance Digit. Humanit. Organ.*

Gladkova, A., Drozd, A., 2016. Intrinsic Evaluations of Word Embeddings: What Can We Do Better? https://doi.org/10.18653/v1/w16-2507.

Received: 17.06.2021 – Accepted: 18.07.2021

Mikolov, T., Chen, K., Corrado, G., Dean, J., 2013. Efficient estimation of word representations in vector space. *1st Int. Conf. Learn. Represent. ICLR* 2013 - Work. Track Proc. 1–12.

Bengio, Y., Ducharme, R., Vincent, P., Jauvin, C., 2003. A Neural Probabilistic Language Model, in: *Journal of Machine Learning Research.* https://doi.org/10.1162/ 153244303322533223.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P., 2011. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*

Pennington, J., Socher, R., Manning, C.D., 2014. GloVe: Global vectors for word representation, in: *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference.* https://doi.org/10.3115/v1/d14 1162.

Bojanowski, P., Grave, E., Joulin, A., Mikolov, T., 2017. Enriching Word Vectors with Subword Information. *Trans. Assoc. Comput. Linguist.* https://doi.org/10.1162/ tacl_a_00051.

Joulin, A., Grave, E., Bojanowski, P., Mikolov, T., 2017. Bag of tricks for efficient text classification, in: *15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017 - Proceedings of Conference.* https://doi.org/10.18653/v1/e17-2068.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., Ruppin, E., 2002. Placing search in context: The concept revisited. *ACM Trans. Inf. Syst.*, 20. https://doi.org/10.1145/503104.503110.

Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., Soroa, A., 2009. A study on similarity and relatedness using distributional and wordnet-based approaches, in: NAACL HLT 2009 - Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, *Proceedings of the Conference.* https://doi.org/10.3115/1620754.1620758.

Miller, G.A., Charles, W.G., 1991. Contextual Correlates of Semantic Similarity. *Lang. Cogn. Process.* 6. https://doi.org/10.1080/01690969108406936.

Rubenstein, H., Goodenough, J.B., 1965. Contextual correlates of synonymy. *Commun. ACM* 8. https://doi.org/10.1145/365628.365657.

Luong, M.T., Socher, R., Manning, C.D., 2013. Better word representations with recursive neural networks for morphology. *CoNLL 2013 - 17th Conf. Comput. Nat. Lang. Learn. Proc.* 104–113.

Bruni, E., Tran, N.K., Baroni, M., 2014. Multimodal distributional semantics. *J. Artif. Intell. Res.* 49. https://doi.org/10.1613/jair.4135.

Radinsky, K., Agichtein, E., Gabrilovich, E., Markovitch, S., 2011. A word at a time: Computing word relatedness using temporal semantic analysis, in: *Proceedings of the 20th International Conference on World Wide Web,* WWW 2011. https://doi.org/10.1145/ 1963405.1963455.

Halawi, G., Dror, G., Gabrilovich, E., Koren, Y., 2012. Large-scale learning of word relatedness with constraints, *in: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* https://doi.org/10.1145/2339530.2339751.

Turney, P.D., 2001. Mining the web for synonyms: PMI-IR versus LSA on TOEFL, *in: Lecture Notes in Computer Science.* https://doi.org/10.1007/3-540-44795-4_42.

Hill, F., Reichart, R., Korhonen, A., 2015. Simlex-999: Evaluating semantic models with (Genuine) similarity estimation. *Comput. Linguist. 41.* https://doi.org/10.1162/COLI_a_00237.

Baker, S., Reichart, R., Korhonen, A., 2014. An unsupervised model for instance level subcategorization acquisition, *in: EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference.* https://doi.org/10.3115/v1/d14-1034.

Gerz, D., Vulic, I., Hill, F., Reichart, R., Korhonen, A., 2016. Simverb-3500: A large-scale evaluation set of verb similarity, in: EMNLP 2016 - Conference on Empirical Methods in Natural Language *Processing, Proceedings.* https://doi.org/10.18653/v1/d16-1235.

Tomas Mikolov, Wen-tau Yih, G.Z., 2013. *Linguistic Regularities in Continuous Space Word Representations* - Microsoft Research, Hlt-Naacl.

Almuhareb A., 2006. *Attributes in lexical acquisition.* Ph.D. thesis, University of Essex.

Baroni, M., Lenci, A., 2011. How we Blessed distributional semantic evaluation, *in: Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics.*

Baroni, M., Murphy, B., Barbu, E., Poesio, M., 2010. Strudel: A corpus-based semantic model based on properties and types. *Cogn. Sci.* 34. https://doi.org/10.1111/j.1551-6709.2009.01068.x.

Jurgens, D.A., Mohammad, S.M., Turney, P.D., Holyoak, K.J., 2012. SemEval-2012 Task 2: Measuring degrees of relational similarity, in: **SEM 2012 - 1st Joint Conference on Lexical and Computational Semantics.*

Menon, M., 2017. Building superlatives from property concept expressions. *Proc. Linguist. Soc. Am.* https://doi.org/10.3765/plsa.v2i0.4086

Baroni, M., Evert, S., Lenci, A., 2008. Lexical semantics: bridging the gap between semantic theory and computational simulations. *Eur. Summer Sch. Logic, Lang. Inf.*