IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Semantic Word Embeddings For Marathi Language

Apurv Choudhari, Ameya Ekbote, Prerona Chaudhuri
Student, Student, Student
Department of Computer Engineering
Vishwakarma Institute of Technology
Pune, India
apurv.choudhari16@vit.edu

***Abstract*—Representing words as vectors which encode their semantic properties is a vital component in natural language processing. Recent advances in distributional semantics have led to the rise of neural network-based models that use unsupervised learning to represent words as dense, distributed vectors, called word embedding. These embeddings have led to breakthroughs in performance in multiple natural language processing applications and hold the key to improving natural language processing for low-resource languages by helping machine learning algorithms learn patterns more easily from these richer representations of words, thereby allowing better generalisation from less data. In this paper, we train the CBOW model on more than 2 million Marathi sentences to create the first large-scale word embedding for the Marathi language.**

We analyse the quality of the learned embedding by looking at the closest neighbours to different words in the vector space and find that they capture a high degree of syntactic and semantic similarity between words. We evaluate this quantitatively by experimenting with two approaches namely training the model without using morphological analysis on the given dataset and by applying morphological analysis.

Keywords— Marathi Word Embeddings, Word Vectors, CBOW, Morphological Analysis, PMI, Skip-gram, Glove

I. INTRODUCTION

Word embeddings are useful for a wide variety of applications beyond NLP such as information retrieval, recommendation, question answer system, search engine and link prediction in knowledge bases, etc, which all have their own task-specific approaches. Wu et al. (2017) propose a general-purpose model that is compatible with many of these applications and can serve as a strong baseline.

This has been done to a great extent for English, but the Internet today does not recognize Marathi for doing the same as in English. There are a lot of scriptures in Marathi, hence we see the potential of doing the same with this language and hence let the Internet reach out to more people.

II. LITERATURE REVIEW

Word Embeddings Methods

Word embeddings models can be divided into main categories:

_ Count-based methods

_ Predictive methods

Models in both categories share, in at least some way, the assumption that words that appear in the same contexts share semantic meaning. One of the most influential early works in count-based methods is the LSI/LS (Latent Semantic Indexing/Analysis method. This hypothesis leads to an amazingly simple albeit a very high-dimensional word embedding. Formally, each word can be represented as a vector in $R\hat{}N$ where N is the unique number of words in each dictionary (in practice N=100,000). Then, by taking a very large corpus (e.g., Wikipedia), let $Count5(w1, w2)$ be the number of times $w1$ and $w2$ occur within a distance 5 of each other in the corpus. Then the word embedding for a word w is a vector of dimension N, with one coordinate for each dictionary word. The coordinate corresponding to word w2 is $Count5(w, w2)$. The problem with the resulting embedding is that it uses extremely high-dimensional vectors. In the LSA article, it was empirically discovered that these embeddings can be reduced to vectors $R300$ by doing a rank-300 SVD on the NxN original embeddings matrix.

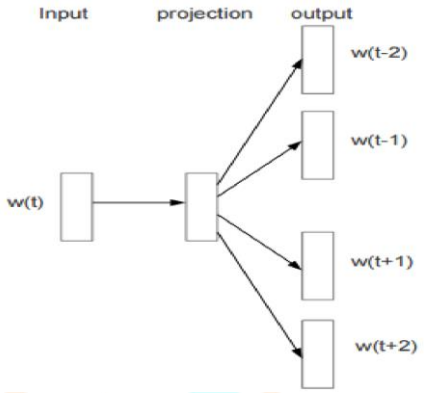
The second family of methods, sometimes also referred as neural probabilistic language models, had theoretical and some practical appearance as early as 1986 [Hinton, 1986]. Unlike count-based models, predictive models try to predict a word from its neighbors in terms of learned small, dense embedding vectors. Two of the most popular methods which appeared recently are the Glove (Global Vectors for Word Representation) method [Pennington et. al., 2014], which is an unsupervised learning method, although not predictive in the common sense,

and Word2Vec, a family of energy based predictive models, presented by [Mikolov et. al., 2013]. As Word2Vec is the embedding method used in our work it shall be briefly discussed here.

Word2Vec:

Word2vec is a particularly computationally efficient predictive model for learning word embeddings from raw text. It comes in two flavors, the Continuous Bag-of-Words model (CBOW) and the Skip-Gram model. Algorithmically, these models are similar, except that CBOW predicts target words (e.g., 'mat') from source context words ('the cat sits on the'), while the skip-gram does the inverse and predicts source context words from the target words.

In the skip-gram model a neural network is trained over a large corpus in where the training objective is to learn word vector representations that are good at predicting the nearby words. The method is also using a simplified version of NCE [Gutmann and HyvÃd'rinen, 2012] called Negative sampling.



The skip-gram model architecture.

This objective enables the model to differentiate data from noise by means of logistic regression, thus learning high-quality vector representations. The CBOW does the same, but the direction is inverted. In other words, the CBOW trains a binary logistic classifier where, given a window of context words, gives a higher probability to "correct" if the next word is correct and a higher probability to "incorrect" if the next word is a random sampled one. Notice that CBOW soothes over a lot of the distributional information (by treating an entire context as one observation). For the most part, this turns out to be a useful thing for smaller datasets. However, skip-gram treats each context-target pair as a new observation, and this tends to do better when we have larger datasets. Finally, the vector we used in our work had a dimension of 300. The Network was trained on the Google News dataset which contains 30 billion training words, with negative sampling as mentioned above. These embeddings can be found online2. A lot of follow-up work was done on the Word2Vec method. The simplest property of embeddings obtained by all the methods described above is that similar words tend to have similar vectors. More formally, the similarity between two words (as rated by humans on a [-1,1] scale) correlates with the cosine similarity between those words' vectors. The fact that words embedding are related to their context words stand behind the similarity property as naturally, similar words tend to appear in similar context. This, however, creates the problem that antonyms (e.g., cold and hot etc.) also appear with the same context while they are, by definition, have opposite meaning. In [Mikolov et. al., 2013] the score of the (accept, reject) pair is 0.73, and the score of (long, short) is 0.71.

The problem of antonyms was tackled directly by [Schwartz et al., 2015]. In this article, the authors introduce a symmetric pattern-based approach to word representation which is particularly suitable for capturing word similarity. Symmetric patterns are a special type of patterns that contain exactly two wildcards and that tend to be instantiated by wildcard pairs such that each member of the pair can take the X or the Y position. For example, the semantically plausible expressions "cats or dogs" and "dogs or cats" exemplify the symmetry of the pattern "X or Y". Specifically, it was found that two patterns are particularly indicative of antonymy - "from X to Y" and "either X or Y".

FRANCE	JESUS	XBOX	REDDISH	SCRATCHED	MEGABITS
454	1973	6909	11724	29869	87025
PERSUADE	THICKETS	DECADENT	WIDESCREEN	ODD	PPA
FAW	SAVARY	DIVO	ANTICA	ANCHIETA	UDDIN
BLACKSTOCK	SYMPATHETIC	VERUS	SHABBY	EMIGRATION	BIOLOGICALLY
GIORGI	JFK	OXIDE	AWE	MARKING	KAYAK
SHAHEED	KHWARAZM	URBINA	THUD	HEUER	MCLARENS
RUMELIA	STATIONERY	EPOS	OCCUPANT	SAMBHAJI	GLADWIN
PLANUM	ILIAS	EGLINTON	REVISED	WORSHIPPERS	CENTRALLY
GOA'ULD	GSNUMBER	EDGING	LEAVENED	RITSUKO	INDONESIA
COLLATION	OPERATOR	FRG	PANDIONIDAE	LIFELESS	MONEO
BACHA	W.J.	NAMSOS	SHIRT	MAHAN	NILGIRIS

What words have embeddings closest to a given word?

Linear analogy relationships:

A more interesting property of recent embeddings [Mikolov et. al., 2013] is that they can solve analogy relationships via linear algebra. This is even though those embeddings are being produced via nonlinear methods. For example, vQueen is the most similar answer to the vKing - vMen + vWomen equation. It turns out, though, that much more sophisticated relationships are also encoded in this way as we can see in figure 5 below.

An interesting theoretical work on non-linear embeddings (especially PMI) was done by [Arora et al., 2015]. In their article they suggest that the creation of a textual corpus is driven by the random walk of a discourse vector $ct^2 < d$, which is a unit vector whose direction in space represents what is being talked about. Each word has a (time-invariant) latent vector $vw^2 < d$ that captures its correlations with the discourse vector. Using a word production model, they predict that words occurring at successive time steps will also tend to have vectors that are close together, thus explaining why similar words have similar vectors. Using the above model, the authors introduce the "RELATIONS = DIRECTIONS" notion for linear analogies. The authors claim that for each relation R, some direction μR can be found which satisfies some equation. This leads to the finding that given enough examples of a relationship R, it is possible to compute μR using SVD and then given a

pair of words with a relation R and a word c, find the best analogy with word d by finding the pair c and d such that $vc - vd$ has highest possible projection over μR. In this way, thay also explain that low dimension of the vectors has a "purifying" effect that reduces the effect of the overfitting coming from the PMI approximation, thus achieving much better results than higher dimensional vectors.

Relationship	Example 1	Example 2	Example 3
France - Paris	Italy: Rome	Japan: Tokyo	Florida: Tallahassee
big - bigger	small: larger	cold: colder	quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France	Berlusconi: Italy	Merkel: Germany	Koizumi: Japan
copper - Cu	zinc: Zn	gold: Au	uranium: plutonium
Berlusconi - Silvio	Sarkozy: Nicolas	Putin: Medvedev	Obama: Barack
Microsoft - Windows	Google: Android	IBM: Linux	Apple: iPhone
Microsoft - Ballmer	Google: Yahoo	IBM: McNealy	Apple: Jobs
Japan - sushi	Germany: bratwurst	France: tapas	USA: pizza

Relationship pairs in a word embedding. From [Mikolov et. al., 2013]

Marathi is the language spoken primarily by the native people of Maharashtra, a state of Indian sub-continent. There are about 90 million people who speak Marathi worldwide. It is one of the oldest of the Indo-Aryan regional languages. It is thought to be approximately 1300 years old, and it is considered that this language evolved from Sanskrit and Prakrit (a group of languages spoken in ancient India), and its syntax and grammar, from Pali. Three Prakrit languages, simpler in structure, emerged from Sanskrit. These were Saurseni, Magadhi and Maharashtri. Maharashtri gradually evolved into Marathi in the 15th and 16th centuries. Marathi is the fourth largest language of India. Outside of India it is spoken in Israel and Mauritius.

English Case	**Singular (एकवचन)**	**Case Ending Used**	**Plural (अनेकवचन)**	**Case Ending Used**
Description				
Nominative case	–	–	विद्यार्थी अभ्यास करतात	–
Accusative case	त्याने संस्थे**स** देणगी दिली	*-sa* (-स)	त्यानी संस्थे**स** देणगी दिली	*-sa* (-स)
Accusative case	घरा**ला** रंग दिला	*-lā* (-ला)	घरां**ना** रंग दिला	*-nā* (-ना)
Instrumental case	विद्यार्थी पेन्सिल**नी** चित्र काढतो	*-nī* (-नी)	विद्यार्थी पेन्सिल**नी** चित्र काढतात	*-nī* (-नी)
Instrumental case	मुलगा दारा**शी** उभा होता	*shī* (-शी)	मुलें दारा**शी** उभी होती	*shī* (-शी)
Dative case	मी मुला**स** ओळखतो	*-sa* (-स)	मी मुलां**ना** ओळखतो	*-nā* (-ना)
Dative case	मी विद्यार्थ्या**ला** ओळखतो	*-lā* (-ला)	मी विद्यार्थ्यां**ना** ओळखतो	*-nā* (-ना)
Ablative case	मुलगा घरू**न** निघाला	*-un* (-उन)	मुलें घरू**न** निघाली	*-un* (-उन)
Ablative case	मुलगा गावा**हुन** आला	*-hun* (हुन)	मुलं गावा**हुन** आली	*-hun* (हुन)
Genitive case	घरा**चा** दरवाजा सुंदर आहे	*cā* (-चा)	घरां**चे** दरवाजे सुंदर आहेत	*-ce* (-चे)
Genitive case	मुलां**ची** तब्येत सुधारली आहे	*-cī* (-ची)	मुलां**च्या** तब्येती सुधारल्या आहेत	*-cyā* (-च्या)
Genitive case	मुला**चे** प्रगती पत्रक मिळाले	*-ce* (-चे)	मुलां**ची** प्रगती पत्रके मिळाली	*-cī* (-ची)
Locative case	मुलगा घरा**त** होता	*-ta* (-त)	मुलें घरा**त** होती	*-ta* (-त)
Locative case	मुलगा घरी होता	*-ī* (-ई)	मुलें घरी होती	*-ī* (-ई)
Locative case	गाय घरी परतली	*-ī* (-ई)	गाई घरा परतल्या	*-ā* (-आ)
Vocative case	–	–	मुला**नो** शांत बसा	*-no* (-नो)

Case:

There are differences of opinion regarding grammatical cases in Marathi. According to one view, there are two cases: direct, which is unmarked (e.g., Ram ('Ram')) and oblique, which is used before postpositions (e.g., ram-a-pasun ('from Ram'), ram-a-la ('to Ram'), '-a' being the case marker and '-la' the dative postposition). According to the alternative analysis, there is a distinction between two classes of "postpositions". Some of them, like -pasun ('from') have a wide range of meanings and can be separated from the noun by clitics like -chya (e.g., ram-a-chya-pasun), while others (like -la) are only used to mark arguments and cannot be separated from the noun by clitics (*ram-a-chya-la is ungrammatical). The latter are then considered to be the case markers. In this view, the cases are nominative (unmarked), accusative/dative (singular -la, plural -na), ergative, which is traditionally called 'instrumental' (sg. -ne, pl. -ni) and genitive/possessive (-tsa, -tse, -tʃa, -tʃi). The class of true postpositions will then include -hatun 'through', -hu(n) 'from'/ablative, -t locative, -gaji 'in place of' and many more. The genitive markers inflect to agree with the governing noun. The form of the oblique suffix depends on the gender and the final vowel of the word it is suffixed to.

Traditional Grammar:

In traditional analyses which follow the pattern of Sanskrit grammatical tradition, case suffixes are referred to as vibhaktī pratyaya (विभक्ति प्रत्यय). There are eight such vibhaktī (विभक्ति) in Marathi. The form of the original word changes when such a suffix is to be attached to the word, and the new, modified root is referred to as saamaanya roop (Original Form) of the original word. For example, the word ghodā (घोडा "horse") gets transformed into ghodyā- (घोड्या-) when the suffix -var (वर- "on") is attached to it to form ghodyāvar (घोड्यावर "on the horse").

III. Solution Approach

We have implemented this project on python using Gensim and Indic NLP library. We also used Google Colab GPU facility for faster training of the model. The model created by Word2Vec largely depends on the dataset that is provided. Hence more the data, the better the

word embedding are created. For the current project we combined three datasets, Marathi news, Marathi Wikipedia and text from Marathi web pages, resulting into a total of around 2M sentences.

The data had to be converted to a matrix format (2 dimensional lists in Python), to be provided as input to the Word2Vec function of Gensim library. Hence, we developed the code for removing all the foreign characters, since our focus was only on Marathi script. We chose not remove words which our foreign to the Marathi language since, we wanted our embedding to be compatible with the current Marathi language spoken. We implemented the code over our local computers which took about 24 hours to complete the preprocessing and save it as an NPY file (.npy). We could also create embedding catering to specific needs, like for example, only trained on dataset for textbooks of school going students, since a word is known by the company of words it keeps…
We have taken the Unicode format range for Marathi script (same as Devnagari script) for collecting the required words \u0900-\u0967.

Marathi language is a morphologically rich language. Due to its dynamic structure of root words and suffixes, it is difficult to be able to write a generalized rule-based approach for the language, unlike English where the root words can be easily extracted. The least that can be done for the morphological analysis is writing rule-based model for the word and its corresponding structure. The same has been achieved to an appreciable extent by the Indic NLP library developed by Anoop Kunchukuttan , IIT B, which has been also used in this project. But even this model seems to have its own limitations like for example

दातांच्या (of teeth) is broken down to

दाता + ंच्या

But we know the actual morphological analysis is

दात + अंच्या

We downloaded the Indic NLP library locally on the computer and set up the required environment for doing the same.

Without Morphological Analysis

The dataset is combination of multiple datasets. This dataset is converted into a huge matrix.

Each line in dataset is processed and a list is created for every line. Each element in list will be a word in the corresponding line. In such a way all the lines in the dataset are converted into lists. All these lists are combined to form a huge matrix. This matrix is given to word2vec function in gensim. CBOW is used to train all the vectors which can be done by setting SG parameter in word2vec function as 0.

The model identifies मुलींना (To the girls) and मुली (Girls) as two different words and hence creates different vectors for both which increases the vocabulary.

With Morphological Analysis

The main idea behind this approach is to develop a vocabulary of root words rather than those words with suffixes. This will lead to better representation of data and words like मुलींना (To the girls) and मुली (Girls) will be mapped to मुली. Hence better results are to be obtained. Moreover, this will lead to a smaller vocabulary than in the previous case. We can also omit the suffixes while preprocessing but we haven't implemented the same owing to overhead of again writing a dictionary for the types of suffixes.

In this method with morphological analysis, we seperate suffixes from words.

Consider the example, "जाणार यानेसुधा" {jaa-naa-ryane-suddhaa } {the one going also (instrumenta l)} {जा + णारा +ने+सुधा}. The root is the verb "जा" {ja a } {go} attached with three suffixes "णारया" {naa rya }, ने{ne

} and "सुधा" {suddha } {also} respectively. Here "णारया" has "ने" as suffix which in turn has "सुधा" as suffix.

Now, the elements in the list will be divided into the suffixes and word. Each element in the list will be either a suffix or a word. All the lists are processed in such a way and matrix is created. This matrix is given as input to the word2vec function which will train all the vectors using CBOW.

IV. RESULTS

We were able to successfully implement the functions of Gensim like most similar, similarity, does not match giving the most similar word for the entered word, similarity between two entered words and finding out the odd one out using cosine similarity. The project stands limited due to lack of proper dataset and difficulty in generalizing the process of morphological analysis. Though a rule-based approach has been done for morphological analysis by Indic NLP, still there are many limitations as discussed in the report.
This project also introduced us to the world of cloud services, since we used GPU provided by Google Colab and trained model within a very short span of time as compared to the time it was taking on our computers. The Word Embedding formed can be used for efficient. Searching, Translation, Tagging, any task related to the processing on the Marathi Language. We analyzed the quality of the learned embedding by looking at the closest neighbors to different words in the vector space and find that they capture a high degree of syntactic and semantic similarity between words. We evaluate this quantitatively by experimenting with two approaches namely training the model without using morphological analysis on the given dataset and by applying morphological analysis. We hope to develop applications by developing better word vectors provided we obtain a better and specific dataset.

Model Parameters:
With morphological analysis:
Vocabulary size: 49890 words
Model type: <class 'gensim.models.word2vec.Word2Vec'>

Parameters of the gensim model are:
Without morphological analysis:
Vocabulary size: 949441 words

V. REFERENCES

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

[2].https://github.com/anoopkunchukuttan/indic_nlp_library

[3].https://radimrehurek.com/gensim/models/word2vec.html

[4]. Urdu Word Embeddings Samar Haider University of Engineering and Technology, Lahore Pakistan samar.haider@kics.edu.pk

[5]. Recent Trends in Deep Learning based Natural Language Processing Bejing institute of China

[6]. Learning Cognitive Features from Gaze data for sentimental sarcasm classification using convolutional Neural Network. IBM research, India.

[7]. Using Word Embeddings for Bilingual Unsupervised WSD. Indian Institute of Technology, Bombay

[8].https://nlp.stanford.edu/pubs/glove.pdf

[9]. https://arxiv.org/pdf/1301.3781.pdf

[10]. https://arxiv.org/pdf/1610.08229.pdf

[11]. https://aclanthology.org/S17-1002.pdf

[12]. https://arxiv.org/abs/1502.03520