NLP Project Presentation

Word Presentation For Paraphrase Myanmar Language Using Word2vec Model

Presented by

Myint Myint Htay (Ph.D-14)

Myat Nyein Chan (Ph.D-11)

May Phyo Aung (Ph.D-7)

Ei Phyu Phyu Mon (Ph.D-9)

28.1.2021

Outline

- Abstract
- Introduction
- Related Work
- Word Segmentation
- Data Collection
- Methodology
- Experimental Setup
- Results and Discussion
- Conclusion

Abstract

- Word2Vec methods are widely used to evaluate similarity scores and to classify text in a variety of language. Both skipgram and CBOW are also two of the most popular methods based on word2vec in state-of-art natural language processing (NLP).
- Skipgram works well with small amount of data and is found to represent rare words well.
- Continuous Bag-of-Word (CBOW) is faster and has better representations for more frequent words.
- This paper analyses performance of two models in similarity scores and processing time for Myanmar language. Data are collected in domain with respect to the traveling and daily general conversations.

Introduction

- Natural Language Processing (NLP) is the field of artificial intelligence that studies the interactions between computers and human languages, in particular how to program computers to process and analyze large amounts of natural language data.
- NLP is often applied for classifying text data. Text classification is the problem of assigning categories to text data according to its content.
- There are different techniques to extract information from raw text data and use it to train a classification model.
- These techniques are:
  - Bag-of-words
  - Word embedding models

  (used with a deep learning neural network such as Word2Vec)
  - State of the art language models

Cont'd

- All of these techniques, we applied this proposed system by using word embedding.
- There is a lot of progress being currently made in NLP using word embedding, it is a positive trend that can be used in a very broad range of practical NLP applications such as computing the similarities between words, using as features in text classification and different natural language tasks such as sentiment analysis.
- The idea of word2vec (word embeddings) originated from the concept of distributed representation of words, it uses a shallow neural network to learn word embeddings and predicts between every word and its context words so words occurring in similar contexts are related.
- That can be done using 2 different approaches:

  1) **Continuous Bag-of- Words**: starting from the context to predict a word.
  2) **Skip-gram**: starting from a single word to predict its context.

Related Work

- **Carl Allen et al. [1]** showed that where embeddings factorise pointwise mutual information (PMI), it is paraphrasing that determines when a linear combination of embeddings equates to that of another word. They derived a probabilistically grounded definition of paraphrasing that they reinterpret as word transformation, a mathematical description of "w_x is to w_y ". From these concepts proved that the existence of linear relationships between W2V-type embeddings.
- **Chenhui Chu et al. [2]** addressed the OOV problem for low resource SMT by paraphrasing with word embeddings and semantic lexicons. They proposed using semantic lexicons including WordNet, FrameNet, and the Paraphrase Database (PPDB) for paraphrasing. In addition, they applied a method to combine these two types of paraphrases, which achieved further improvements in SMT.

Cont'd

- **David Guthrie et al. [3]** examined the use of skip-grams to overcome the data sparsity problem. NLP researchers investigated skip-gram modeling using one to four skips with various amount of training data and test against similar documents as well as documents generated from a machine translation system. In the paper, they also determined the amount of extra training data required to achieve skip-gram coverage using standard adjacent trigram. These paper was focus to quantify the impact skip-gram modeling has on the coverage of trigram in real text and compared this to coverage obtained by increasing the size of corpus used to build a traditional language model.
- **David Guthrie et al. [4]** in this suvery, authors highlighted the latest studies on using the Word2vec model for sentiment analysis. According to this literature that most studies were used the two methods of word2vec: CBOW and skip-gram, and compared the results from each method. Skip-gram is better for infrequent words than CBOW, however, CBOW is faster and works well with frequent words. Many of the studies in literature applied word2vec using tools such as word2vec tool and FastText.

Cont'd

- **Aye Myat Mon and Khin Mar Soe [5]**, this paper tries to extract the analogous words between Myanmar news articles focus on the bag of words (CBOW) model using different features vector sizes. By analyzing word embedding model are obtained the better results with high-dimensional vectors than low-dimensional vectors to cluster the words based on its relatedness.
- **Hay Mar Su Aung and Win Pa Pa [6]**, using the concept of word embeddings is to increase the accuracy of the sentiment identification. Word2Vec is used to train for producing high-dimensional word vectors that learns the syntactic and semantic of word. The resulting word vectors train Machine Learning algorithms in the form of classifiers for sentiment identification. The use of word embeddings from the collected real-world datasets improves the accuracy of sentiments classification.

Word Segmentation

- Word segmentation is the very important method for the text analysis level.
- The under resource languages such as Burmese text are not usually separate with white space between words.
- The white spaces are often used to distinguish sentences for easier reading.
- We also used word segmentation method and additionally manually segmented for Burmese word segmentation process.

Data Collection

- In Burmese, various words and various conversation styles for the same performance in **daily conversation** and traveling **domain**.
- Some of the paraphrase sentences are different only one word in that sentence and some sentences are quite different for the whole sentences.
- Some of the sentences are collected from social media (Facebook comments) and the comments are collected from the famous Myanmar news websites by extracting the Facepager Tool (version 4.2.7).
- Comments in social media are collected as needed as the sentences for our requirements in data in the range of travel domain and daily general conversations.
- Moreover, some are collected from the Burmese Wiktionary site and extraction with Web Scraper tool.
- We also used travel domain data from.
- Using these words and we built more of the paraphrase sentences for our research.
- Based on in this research, we used **242,327 total sentences** to train, validate and predict results unsupervisedly.
- The Burmese corpus is a UTF-8 plain text file.

Word Embedding

- Word embeddings are basically a form of word representation that bridges the human understanding of language to that of a machine.
- They have learned representations of text in an n-dimensional space where words that have the same meaning have a similar representation.
- Meaning that two similar words are represented by almost similar vectors that are very closely placed in a vector space. These are essential for solving most natural language processing problems.
- Thus when using word embedding, all individual words are represented as real-valued vectors in a predefined vector space.
- Each word is mapped to one vector and the vector values are learned in a way that resembles a neural network.
- As the machine learning models cannot process text so we need to figure out a way to convert these textual data into numerical data.

What is Word2vec?

- Word2vec is a method to efficiently create word embeddings by using a two-layer neural network.
- It was developed to make the neural-network-based training of the embedding more efficient and then has become the de facto standard, for developing pre-trained word embedding.
- The input of word2vec is a text corpus and its output is a set of vectors known as feature vectors that represent words in that corpus.
- While Word2vec is not a deep neural network, it turns text into a numerical form that deep neural networks can understand.
- The Word2Vec objective function causes the words that have a similar context to have similar embeddings. Thus in this vector space, these words are really close.

Continuous Bag-of-Words Model (CBOW)

- CBOW predicts the probability of a word to occur given the words surrounding it, and can consider a single word or a group of words.
- But for simplicity, a single context word will be taken and try to predict a single target word.
- For example, "သူ ကျောင်း သို့ သွားသည်"
  - First, we convert each word into a one-hot encoding form. Also, we'll not consider all the words in the sentence but only take certain words that are in a window.
  - The middle word is to be predicted and the surrounding two words are fed into the neural network as context. The window is then slid and the process is repeated again.
  - Finally, after training the network repeatedly by sliding the window a shown in Fig. 1b, we get weights which we use to get the embedding as shown in Fig. 1c.

Model architectures of CBOW

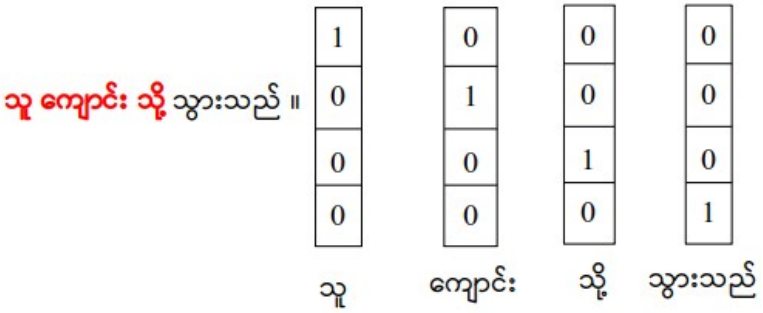

Fig 1: (a) Convert each word into a one-hot encoding form

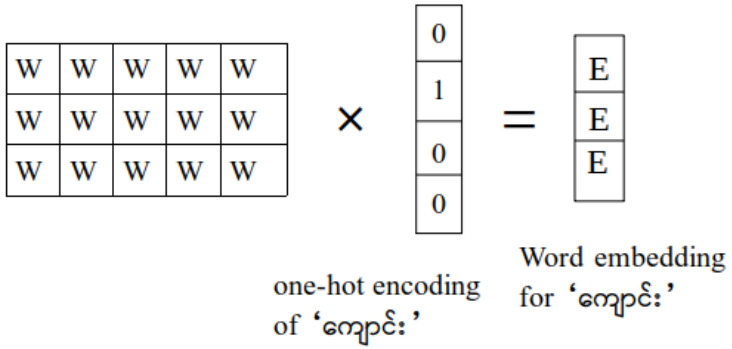

Fig 1: (b) Weights are used to get the embedding

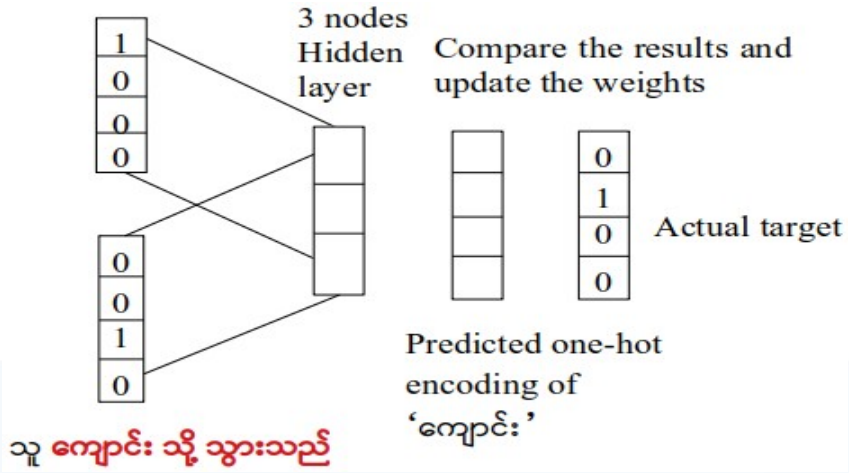

Fig 1: (c) Training the network repeatedly by sliding window

Skip-gram Model

- The Skip-gram model architecture usually tries to achieve the reverse of what the CBOW model does.
- It tries to predict the source context words (surrounding words) given a target word (the center word).
- The working of the skip-gram model is quite similar to the CBOW but there is just a difference in the architecture of its neural network.

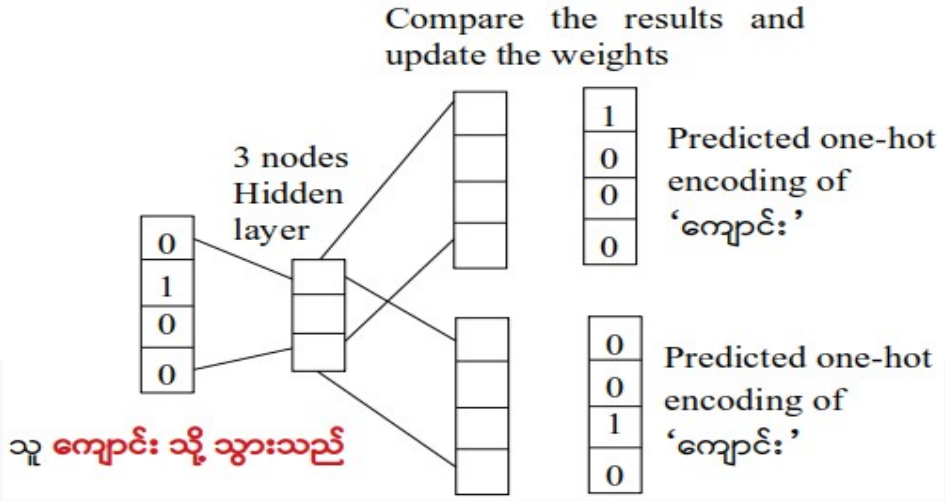

Fig 2: Training the network repeatedly by sliding the window

Experimental Setup

- In this proposed paper, we applied fastText python library (fasttext 0.9.2). It is a library for efficient learning of word representations and sentence classification.
- fastText provides two models for computing word representations: skipgram and CBOW (’continuous-bag-of-words’).
- Let us illustrate this difference with an example: given the sentence ’Poets have been mysteriously silent on the subject of cheese’ and the target word ’silent’, a skipgram model tries to predict the target using a random close-by word, like ’subject’ or ’mysteriously’.
- The CBOW model takes all the words in a surrounding window, like been, mysteriously, on, the, and uses the sum of their vectors to predict the target.
- In practice, we observed that skipgram models works better with subword information than CBOW.
- We run fastText with the default parameters, but depending on the data, these parameters may not be optimal.

Cont'd

- In introduction to some of the key parameters for word vectors, the most important parameters of the model are its dimension and the range of size for the subwords.
- The dimension (dim) controls the size of the vectors, the larger they are the more information they can capture but requires more data to be learned. But, if they are too large, they are harder and slower to train.
- By default, we use 100 dimensions, but any value in the 100-300 range is as popular.
- The subwords are all the substrings contained in a word between the minimum size (minn) and the maximal size (maxn).
- By default, we take all the subword between 3 and 6 characters, but other range could be more appropriate to different languages.
- Depending on the quantity of data you have, you may want to change the parameters of the training.

Cont'd

- In this paper, the epoch, learning rate and thread used by default.
- In order to find nearest neighbors, we need to compute a similarity score between words.
- Our words are represented by continuous word vectors and we can thus apply simple similarities to them.
- In a similar spirit, one can play around with word analogies by applying (gensim 3.7.3) to find similarity words.
- Gensim is a Python library for topic modelling, document indexing and similarity retrieval with large corpora.
- Target audience is the natural language processing (NLP) and information retrieval (IR) community.

Result and Discussion

- At first, we experimented both skipgram and CBOW based on only training data 100 sentences.
- The experiment of skipgram, we can see if our model can guess what is to အဆာပြေ, and what အဆာပြေ is to မနက်စာ. This can be done with the analogies functionality. It takes a word triplet (like မနက်စာ ထမင်း အဆာပြေ) and outputs the analogy as:

Query triplet (A - B + C)? မနက်စာ - ထမင်း + အဆာပြေ

ချင်ရဲပြေ 0.745777
ထမင်းအေး 0.720301
အာသာပြေ 0.719034
ထမင်းဟင်း 0.714321
ထမင်းမေ့ဟင်းမေ့ 0.69706
ဆာ 0.692757
ထမင်းရေ 0.681634
စားနေကျ 0.680725
အာသီသ 0.678317
အာသာဆန္ဒ 0.677543

Cont'd

- In the experiment of CBOW, we can see if our model can guess what is to အဆာပြေ, and what အဆာပြေ is to မနက်စာ. This can be done with the analogies functionality. It takes a word triplet like (မနက်စာ ထမင်း အဆာပြေ) and outputs the analogy as:

Query triplet (A - B + C)?မနက်စာ - ထမင်း + အဆာပြေ

နေ 0.148211
ပါ 0.142243
မယ် 0.13875
ကြ 0.128288
စောက် 0.128086
ရှိ 0.127845
ရ 0.12631
နိုင် 0.104394
တွေ 0.066768
။ 0.0393234

Cont'd

- And then, we analyzed both skipgram and CBOW based on training data (242,327 sentences).

Query triplet (A - B + C)? မနက်စာ - ထမင်း + အဆာပြေ

Skipgram Output:

ထမင်းဟင်း 0.743704
ထည့်သွင်း 0.682644
ဒင်း 0.67533
မကျန်မကြွင်း 0.672617
ထမင်းရည်ပူလာလျှာလွဲ 0.672009
ဒံပေါက်ထမင်း 0.657037
မဂ္ဂင်း 0.657002
ဟိုသင်း 0.648797
ထမင်းစားပြီး 0.64839
ည့်မြေ 0.647002

CBOW Output:

ထမင့်းဟင့်း 0.777087
လမင်း 0.749024
အိုမင်း 0.731996
မယ်မင်း 0.718184
မကျန်မကြွင်း 0.712241
မဂ္ဂင်း 0.710967
စင်း 0.708923
အခင်း 0.708785
ဒင်း 0.707475
မနက်လင်း 0.70745

Cont'd

- Besides, we studied on the experiment of subword for both skipgram and CBOW. Using subword-level information is particularly interesting to build vectors for unknown words. For skipgram model, by using subword ('အဆာပြေ') information gives the following list of nearest neighbors:

Query word? အဆာပြေ

အမောပြေ 0.846186
အာသာပြေ 0.829208
ညာပြော 0.770294
အဆင်ပြေပြေ 0.770002
စိမ်ပြေနပြေ 0.756734
ဘာပြော 0.750311
ခရာတာတာပြော 0.744353
ဆိုကျပြီမလား: 0.740593
ပြေစာ 0.738888
ပြေ 0.738307

Cont.

- For CBOW model, by using subword (‘အဆာပြေ’) information gives the following list of nearest neighbors:

Query word? အဆာပြေ

အဆင်ပြေပြေ 0.883487
အာသာပြေ 0.87291
ဆင်ပြေ 0.848755
အဆင်ပြေ 0.823117
စိမ်ပြေနပြေ 0.818285
အမောပြေ 0.790669
အဆင်မပြေ 0.790384
ဖျန်ဖြေ 0.779151
ညာပြော 0.768855
ခရာတာတာပြော 0.767131

Conclusion

- Performance of two word2vec methods for myanmar language are evaluated in similarity scores and processing time.
- Data about travel domain and daily general conversations are collected from social media.
- For both CBOW and skipgram, sentences are evaluated with style of nearess and subword.
- CBOW achieved better similarity scores and faster than skipgram in the experiment of data size (242,327 sentences).
- But skipgram also outperforms good result in the experiment of data size only (100 sentences).
- Consequently, the quality of the analogies depend on the dataset used to train the model and one can only hope to cover fields only in the dataset.
- As our future work, the strength of CBOW and skipgram methods based on word2vec will be applied for text classification and sentiment analysis for Myanmar NLP trend.

Thank You!