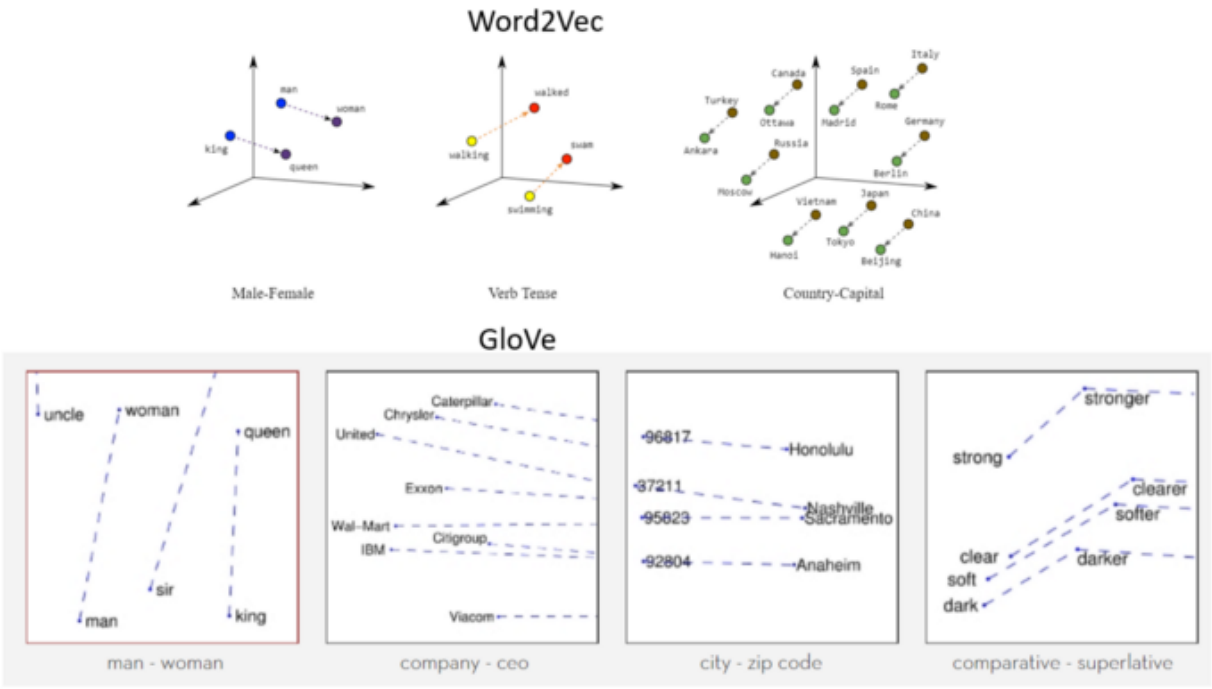
Word Embedding

What exactly are embeddings and how are they utilised in text processing?



The term Natural Language Processing(NLP) refers to computer systems that can read human language and translate it into other languages. An attempt is made by NLP to extract information from sentences in human language, such as English or Hindi. NLP can be used for a variety of tasks, some of which are listed below.

- Text summarization: extractive or abstractive text summarization
- Sentiment Analysis
- Neural machine translation: translating from one language to another
- Chatbots

How can we transform text to numbers for use in machine learning and deep learning algorithms?

Bag of words(BOW)

Features can be extracted from text using the Bag of Words technique, which is straightforward and widely used. This technique uses a bag of words to count how many times each word appears in the text. Vectorization is another name for this process.

Steps for creating BOW

- Tokenize the text into sentences
- Tokenize sentences into words
- Remove punctuation or stop words
- Convert the words to lower text
- Create the frequency distribution of words

For more information regarding Bag of Words refer to the previous article.

What is the problem with the bag of words?

Each document is represented as a word-count vector in the bag of words model. These totals can be binary, meaning a word appears in the text or it doesn't, or they can be absolute, meaning the word appears or it doesn't. The vector's length is determined by the vocabulary's length. The bag of words will be a sparse matrix if the majority of the components are zero.

We'd have a sparse matrix in deep learning because we'll be working with a lot of training data. It's more difficult to model sparse representations, both computationally and informationally.

Huge amount of weights: There are a large number of weights in a neural network when the input vectors are large.

Computationally intensive: Adding weights increases the amount of computation needed to train and predict.

A collection of words that exist in the text or sentences with the word counts but do not have any significant relationships and do not take word order into account. The sequence of the words in a bag is completely irrelevant.

There is a solution to these issues with Word Embedding.

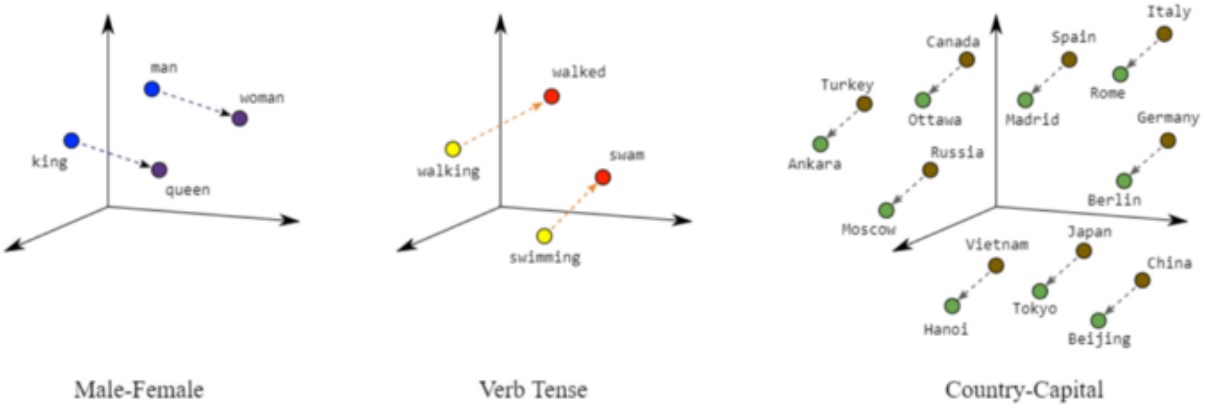
Large sparse vectors are translated into a lower-dimensional space using embeddings, and semantic links are preserved.

When using word embeddings, each word in a domain or language is represented as a lower-dimensional real-valued vector.

Data mapped from a high-dimensional to a low-dimensional space solves the Sparse Matrix problem with BOW

Solving BOW's relationship concerns involves clustering vectors of semantically related objects adjacent to one another in close proximity As can be seen in the illustration below, words with comparable meaning have similar distances in the vector space.

As well as the verb tense and country and its capitals, which are recorded in vector space to preserve the semantic linkages, the king is to the queen as a man is to a woman.



Male-Female Verb Tense Country-Capital

How are semantically similar items placed close to each other?

Let's utilise recommendation engines as an example of collaborative filtering to demonstrate.

Users' prior purchases of other users with comparable interests are used by recommendation algorithms to anticipate what they will buy. making use of filtering that is collaborative

Recommendation engines are used by Amazon and Netflix to make suggestions to their customers about products or movies.

In collaborative filtering, all of the identical products purchased by numerous customers are merged into a single low-dimensional space. Collaborative filtering. The nearest neighbourhood algorithm refers to this low-dimensional space because it contains similar products close together.

The nearest neighbourhood strategy is employed for the purpose of grouping together items that have semantically comparable meanings.

What is the best way to reduce the dimensionality of high-dimensional data?

Using standard Dimensionality reduction techniques

Word embeddings can be created using standard dimensionality reduction techniques like Principal Component Analysis (PCA). Through the use of the BOW, PCA seeks out dimensions with high correlation so that they can be collapsed into a single dimension.

Word2Vec

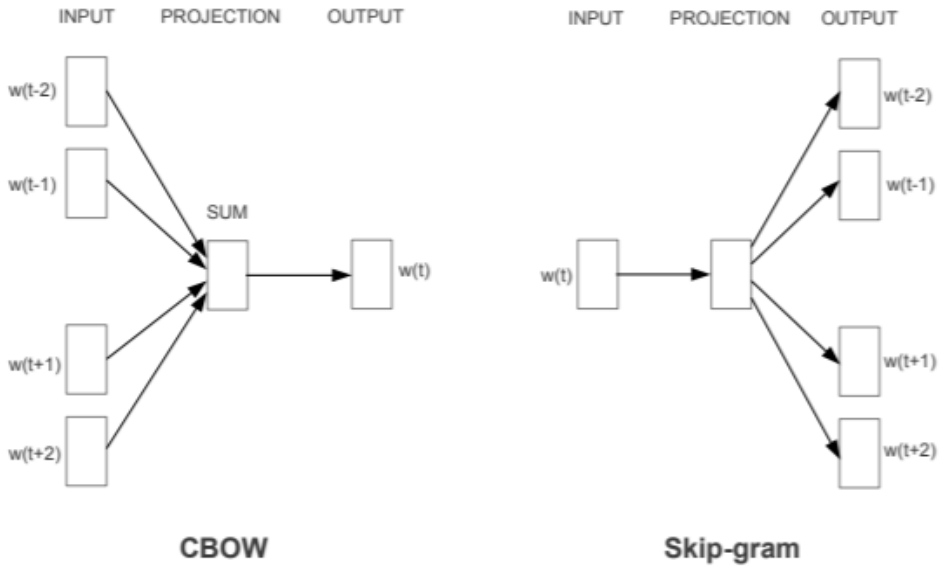
Word2vec is a Google-developed method used to develop word embeddings. word2vec is based on the notion that data is distributed randomly. Distributional hypotheses claim that similar-sounding surrounding words are more likely to have comparable meanings. Using this, we can map comparable phrases to vectors that have geometrically near embeddings.

Instead of using a CBOW or skip grammes, the distributional hypothesis employs a continuous bag of words (CBOW).

For the sake of this tutorial, we'll use Word2vec models. These shallow neural networks include three layers: an input, a projection, and an output. As a result of its training, it has become proficient at reconstructing spoken discourse situations. Word2vec neural network's input layer utilises a bigger corpus of text to create a vector space with many more dimensions, usually in the hundreds. Each word in the text corpus has an associated vector in the space, which is used to identify it.

As a result of using continuous distributed representation of the context, this architecture is known as CBOW (continuous bag of words). It takes into account the past and the future in terms of word order.

This makes it easier to locate similar vectors in the corpus in the vector space.



A log-linear classifier with continuous projection layer inputs each current word as an input and predicts words before and after the current word using skip gramme instead of predicting the current word on the basis of context.

GloVe: Global Vector for word representation

Pennington and colleagues at Stanford came up with GloVe. Global Vectors gets their name from the fact that the model directly captures information from the entire corpus.

It takes advantage of both of these factors.

- Latent semantic analysis (LSA) is a method for constructing low-dimensional word representations via global matrix factorization
- Methods that use a local context window, such as Mikolov et al skip-gram's model

A sub-optimal vector space structure is indicated by LSA's efficient use of statistical information yet its performance on word analogy is sub-par.

However, methods such as skip-gram, which are not trained on global co-occurrence counts, perform better on the analogy task but use the corpus statistics poorly. GloVe makes efficient use of statistics by training on global word co-occurrence counts using a special weighted least squares model.

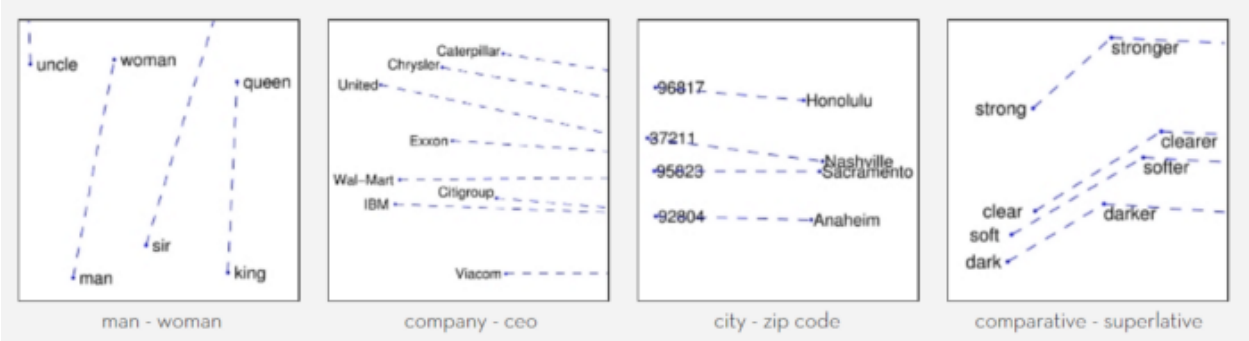
The words i=ice and j=steam in the Thermodynamics domain represent ice and steam, respectively. For each word in this list, the co-occurrence probability ratio can be calculated using various probe words (k).

$$\frac{P_{ik}}{P_{jk}}$$

The ratio should be close to one when looking for words like water or fashion that are either associated with ice or steam, but not both. Words like solid, ice, and water will have a high correlation coefficient, while steam will not.

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

The ratio can distinguish between relevant words (solid and gas) and irrelevant words (water and fashion) better than raw probabilities, and it can also distinguish between the two relevant terms.



Like word pairs like king and queen or brother and sister, gender separates men from women. Mathematically, we may say that the disparities between men and women, kings and queens, brothers and sisters are all equal. GloVe's visualisations show this trait as well as other interesting patterns.