Title: Word embedding

URL: https://en.wikipedia.org/wiki/Word_embedding

PageID: 43561218

Categories: Category:Artificial neural networks, Category:Computational linguistics, Category:Language modeling, Category:Natural language processing, Category:Semantic relations

-----

Supervised learning

Unsupervised learning

Semi-supervised learning

Self-supervised learning

Reinforcement learning

Meta-learning

Online learning

Batch learning

Curriculum learning

Rule-based learning

Neuro-symbolic AI

Neuromorphic engineering

Quantum machine learning

Classification

Generative modeling

Regression

Clustering

Dimensionality reduction

Density estimation

Anomaly detection

Data cleaning

AutoML

Association rules

Semantic analysis

Structured prediction

Feature engineering

Feature learning

Learning to rank

Grammar induction

Ontology learning

Multimodal learning

Apprenticeship learning
Decision trees
Ensembles Bagging Boosting Random forest
Bagging
Boosting
Random forest
k -NN
Linear regression
Naive Bayes
Artificial neural networks
Logistic regression
Perceptron
Relevance vector machine (RVM)
Support vector machine (SVM)
BIRCH
CURE
Hierarchical
k -means
Fuzzy
Expectation–maximization (EM)
DBSCAN
OPTICS
Mean shift
Factor analysis
CCA
ICA
LDA
NMF
PCA
PGD
t-SNE
SDL
Graphical models Bayes net Conditional random field Hidden Markov
Bayes net
Conditional random field
Hidden Markov
RANSAC
k -NN

Local outlier factor

Isolation forest

Autoencoder

Deep learning

Feedforward neural network

Recurrent neural network LSTM GRU ESN reservoir computing

LSTM

GRU

ESN

reservoir computing

Boltzmann machine Restricted

Restricted

GAN

Diffusion model

SOM

Convolutional neural network U-Net LeNet AlexNet DeepDream

U-Net

LeNet

AlexNet

DeepDream

Neural field Neural radiance field Physics-informed neural networks

Neural radiance field

Physics-informed neural networks

Transformer Vision

Vision

Mamba

Spiking neural network

Memtransistor

Electrochemical RAM (ECRAM)

Q-learning

Policy gradient

SARSA

Temporal difference (TD)

Multi-agent Self-play

Self-play

Active learning

Crowdsourcing

Human-in-the-loop

Mechanistic interpretability

RLHF

Coefficient of determination

Confusion matrix

Learning curve

ROC curve

Kernel machines

Bias–variance tradeoff

Computational learning theory

Empirical risk minimization

Occam learning

PAC learning

Statistical learning

VC theory

Topological deep learning

AAAI

ECML PKDD

NeurIPS

ICML

ICLR

IJCAI

ML

JMLR

Glossary of artificial intelligence

List of datasets for machine-learning research List of datasets in computer vision and image processing

List of datasets in computer vision and image processing

Outline of machine learning

v

t

e

In natural language processing , a word embedding is a representation of a word. The embedding is used in text analysis . Typically, the representation is a real-valued vector that encodes the meaning of the word in such a way that the words that are closer in the vector space are expected to be similar in meaning. [1] Word embeddings can be obtained using language modeling and feature learning techniques, where words or phrases from the vocabulary are mapped to vectors of real numbers .

Methods to generate this mapping include neural networks , [2] dimensionality reduction on the word co-occurrence matrix , [3] [4] [5] probabilistic models, [6] explainable knowledge base method, [7] and explicit representation in terms of the context in which words appear. [8]

Word and phrase embeddings, when used as the underlying input representation, have been shown to boost the performance in NLP tasks such as syntactic parsing [9] and sentiment analysis . [10]

Development and history of the approach

In distributional semantics , a quantitative methodological approach for understanding meaning in observed language, word embeddings or semantic feature space models have been used as a knowledge representation for some time. [11] Such models aim to quantify and categorize semantic similarities between linguistic items based on their distributional properties in large samples of language data. The underlying idea that "a word is characterized by the company it keeps" was proposed in a 1957 article by John Rupert Firth , [12] but also has roots in the contemporaneous work on search systems [13] and in cognitive psychology. [14]

The notion of a semantic space with lexical items (words or multi-word terms) represented as vectors or embeddings is based on the computational challenges of capturing distributional characteristics and using them for practical application to measure similarity between words, phrases, or entire documents. The first generation of semantic space models is the vector space model for information retrieval. [15] [16] [17] Such vector space models for words and their distributional data implemented in their simplest form results in a very sparse vector space of high dimensionality (cf. curse of dimensionality). Reducing the number of dimensions using linear algebraic methods such as singular value decomposition then led to the introduction of latent semantic analysis in the late 1980s and the random indexing approach for collecting word co-occurrence contexts. [18] [19] [20] [21] In 2000, Bengio et al. provided in a series of papers titled "Neural probabilistic language models" to reduce the high dimensionality of word representations in contexts by "learning a distributed representation for words". [22] [23] [24]

A study published in NeurIPS (NIPS) 2002 introduced the use of both word and document embeddings applying the method of kernel CCA to bilingual (and multi-lingual) corpora, also providing an early example of self-supervised learning of word embeddings. [25]

Word embeddings come in two different styles, one in which words are expressed as vectors of co-occurring words, and another in which words are expressed as vectors of linguistic contexts in which the words occur; these different styles are studied in Lavelli et al., 2004. [26] Roweis and Saul published in Science how to use " locally linear embedding " (LLE) to discover representations of high dimensional data structures. [27] Most new word embedding techniques after about 2005 rely on a neural network architecture instead of more probabilistic and algebraic models, after foundational work done by Yoshua Bengio [28] [circular reference] and colleagues. [29] [30]

The approach has been adopted by many research groups after theoretical advances in 2010 had been made on the quality of vectors and the training speed of the model, as well as after hardware advances allowed for a broader parameter space to be explored profitably. In 2013, a team at Google led by Tomas Mikolov created word2vec , a word embedding toolkit that can train vector space models faster than previous approaches. The word2vec approach has been widely used in experimentation and was instrumental in raising interest for word embeddings as a technology, moving the research strand out of specialised research into broader experimentation and eventually paving the way for practical application. [31]

Polysemy and homonymy

Historically, one of the main limitations of static word embeddings or word vector space models is that words with multiple meanings are conflated into a single representation (a single vector in the semantic space). In other words, polysemy and homonymy are not handled properly. For example, in the sentence "The club I tried yesterday was great!", it is not clear if the term club is related to the word sense of a club sandwich , clubhouse , golf club , or any other sense that club might have. The necessity to accommodate multiple meanings per word in different vectors (multi-sense embeddings) is the motivation for several contributions in NLP to split single-sense embeddings into multi-sense ones. [32] [33]

Most approaches that produce multi-sense embeddings can be divided into two main categories for their word sense representation, i.e., unsupervised and knowledge-based. [34] Based on word2vec skip-gram, Multi-Sense Skip-Gram (MSSG) [35] performs word-sense discrimination and embedding simultaneously, improving its training time, while assuming a specific number of senses for each word. In the Non-Parametric Multi-Sense Skip-Gram (NP-MSSG) this number can vary depending on each word. Combining the prior knowledge of lexical databases (e.g., WordNet , ConceptNet , BabelNet), word embeddings and word sense disambiguation , Most Suitable Sense Annotation (MSSA) [36] labels word-senses through an unsupervised and knowledge-based approach, considering a word's context in a pre-defined sliding window. Once the words are disambiguated, they can be used in a standard word embeddings technique, so multi-sense embeddings are produced. MSSA architecture allows the disambiguation and annotation process to be performed recurrently in a self-improving manner. [37]

The use of multi-sense embeddings is known to improve performance in several NLP tasks, such as part-of-speech tagging , semantic relation identification, semantic relatedness , named entity recognition and sentiment analysis. [38] [39]

As of the late 2010s, contextually-meaningful embeddings such as ELMo and BERT have been developed. [40] Unlike static word embeddings, these embeddings are at the token-level, in that each occurrence of a word has its own embedding. These embeddings better reflect the multi-sense nature of words, because occurrences of a word in similar contexts are situated in similar regions of BERT’s embedding space. [41] [42]

For biological sequences: BioVectors

Word embeddings for n- grams in biological sequences (e.g. DNA, RNA, and Proteins) for bioinformatics applications have been proposed by Asgari and Mofrad. [43] Named bio-vectors (BioVec) to refer to biological sequences in general with protein-vectors (ProtVec) for proteins (amino-acid sequences) and gene-vectors (GeneVec) for gene sequences, this representation can be widely used in applications of deep learning in proteomics and genomics . The results presented by Asgari and Mofrad [43] suggest that BioVectors can characterize biological sequences in terms of biochemical and biophysical interpretations of the underlying patterns.

Game design

Word embeddings with applications in game design have been proposed by Rabii and Cook [44] as a way to discover emergent gameplay using logs of gameplay data. The process requires transcribing actions that occur during a game within a formal language and then using the resulting text to create word embeddings. The results presented by Rabii and Cook [44] suggest that the resulting vectors can capture expert knowledge about games like chess that are not explicitly stated in the game's rules.

Sentence embeddings

The idea has been extended to embeddings of entire sentences or even documents, e.g. in the form of the thought vectors concept. In 2015, some researchers suggested "skip-thought vectors" as a means to improve the quality of machine translation . [45] A more recent and popular approach for representing sentences is Sentence-BERT, or SentenceTransformers, which modifies pre-trained BERT with the use of siamese and triplet network structures. [46]

Software

Software for training and using word embeddings includes Tomáš Mikolov 's Word2vec , Stanford University's GloVe , [47] GN-GloVe, [48] Flair embeddings, [38] AllenNLP's ELMo , [49] BERT , [50] fastText , Gensim , [51] Indra, [52] and Deeplearning4j . Principal Component Analysis (PCA) and T-Distributed Stochastic Neighbour Embedding (t-SNE) are both used to reduce the dimensionality of word vector spaces and visualize word embeddings and clusters . [53]

Examples of application

For instance, the fastText is also used to calculate word embeddings for text corpora in Sketch Engine that are available online. [54]

Ethical implications

Word embeddings may contain the biases and stereotypes contained in the trained dataset, as Bolukbasi et al. points out in the 2016 paper "Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings" that a publicly available (and popular) word2vec embedding trained on Google News texts (a commonly used data corpus), which consists of text written by professional journalists, still shows disproportionate word associations reflecting gender and racial biases when extracting word analogies. [55] For example, one of the analogies generated using the aforementioned word embedding is "man is to computer programmer as woman is to homemaker". [56] [57]

Research done by Jieyu Zhou et al. shows that the applications of these trained word embeddings without careful oversight likely perpetuates existing bias in society, which is introduced through unaltered training data. Furthermore, word embeddings can even amplify these biases. [58] [59]

See also

Embedding (machine learning)

Brown clustering

Distributional–relational database

References

v

t

e

AI-complete

Bag-of-words

n -gram Bigram Trigram

Bigram

Trigram

Computational linguistics

Natural language understanding

Stop words

Text processing

Argument mining

Collocation extraction

Concept mining

Coreference resolution

Deep linguistic processing

Distant reading

Information extraction

Named-entity recognition

Ontology learning

Parsing Semantic parsing Syntactic parsing

Semantic parsing

Syntactic parsing

Part-of-speech tagging

Semantic analysis
Semantic role labeling
Semantic decomposition
Semantic similarity
Sentiment analysis
Terminology extraction
Text mining
Textual entailment
Truecasing
Word-sense disambiguation
Word-sense induction
Compound-term processing
Lemmatisation
Lexical analysis
Text chunking
Stemming
Sentence segmentation
Word segmentation
Multi-document summarization
Sentence extraction
Text simplification
Computer-assisted
Example-based
Rule-based
Statistical
Transfer-based
Neural
BERT
Document-term matrix
Explicit semantic analysis
fastText
GloVe
Language model (large)
Latent semantic analysis
Seq2seq
Word embedding
Word2vec
Corpus linguistics

Lexical resource
Linguistic Linked Open Data
Machine-readable dictionary
Parallel text
PropBank
Semantic network
Simple Knowledge Organization System
Speech corpus
Text corpus
Thesaurus (information retrieval)
Treebank
Universal Dependencies
BabelNet
Bank of English
DBpedia
FrameNet
Google Ngram Viewer
UBY
WordNet
Wikidata
Speech recognition
Speech segmentation
Speech synthesis
Natural language generation
Optical character recognition
Document classification
Latent Dirichlet allocation
Pachinko allocation
Automated essay scoring
Concordancer
Grammar checker
Predictive text
Pronunciation assessment
Spell checker
Chatbot
Interactive fiction
Question answering
Virtual assistant

Voice user interface
Formal semantics
Hallucination
Natural Language Toolkit
spaCy
v
t
e
History timeline
timeline
Companies
Projects
Parameter Hyperparameter
Hyperparameter
Loss functions
Regression Bias–variance tradeoff Double descent Overfitting
Bias–variance tradeoff
Double descent
Overfitting
Clustering
Gradient descent SGD Quasi-Newton method Conjugate gradient method
SGD
Quasi-Newton method
Conjugate gradient method
Backpropagation
Attention
Convolution
Normalization Batchnorm
Batchnorm
Activation Softmax Sigmoid Rectifier
Softmax
Sigmoid
Rectifier
Gating
Weight initialization
Regularization
Datasets Augmentation
Augmentation

Prompt engineering

Reinforcement learning Q-learning SARSA Imitation Policy gradient

Q-learning

SARSA

Imitation

Policy gradient

Diffusion

Latent diffusion model

Autoregression

Adversary

RAG

Uncanny valley

RLHF

Self-supervised learning

Reflection

Recursive self-improvement

Hallucination

Word embedding

Vibe coding

Machine learning In-context learning

In-context learning

Artificial neural network Deep learning

Deep learning

Language model Large language model NMT

Large language model

NMT

Reasoning language model

Model Context Protocol

Intelligent agent

Artificial human companion

Humanity's Last Exam

Artificial general intelligence (AGI)

AlexNet

WaveNet

Human image synthesis

HWR

OCR

Computer vision

Speech synthesis 15.ai ElevenLabs

15.ai

ElevenLabs

Speech recognition Whisper

Whisper

Facial recognition

AlphaFold

Text-to-image models Aurora DALL-E Firefly Flux Ideogram Imagen Midjourney Recraft Stable Diffusion

Aurora

DALL-E

Firefly

Flux

Ideogram

Imagen

Midjourney

Recraft

Stable Diffusion

Text-to-video models Dream Machine Runway Gen Hailuo AI Kling Sora Veo

Dream Machine

Runway Gen

Hailuo AI

Kling

Sora

Veo

Music generation Riffusion Suno AI Udio

Riffusion

Suno AI

Udio

Word2vec

Seq2seq

GloVe

BERT

T5

Llama

Chinchilla AI

PaLM

GPT 1 2 3 J ChatGPT 4 4o o1 o3 4.5 4.1 o4-mini 5

1

2

3

J

ChatGPT

4

4o

o1

o3

4.5

4.1

o4-mini

5

Claude

Gemini Gemini (language model) Gemma

Gemini (language model)

Gemma

Grok

LaMDA

BLOOM

DBRX

Project Debater

IBM Watson

IBM Watsonx

Granite

PanGu-Σ

DeepSeek

Qwen

AlphaGo

AlphaZero

OpenAI Five

Self-driving car

MuZero

Action selection AutoGPT

AutoGPT

Robot control

Alan Turing

Warren Sturgis McCulloch

Walter Pitts

John von Neumann

Claude Shannon

Shun'ichi Amari

Kunihiko Fukushima

Takeo Kanade

Marvin Minsky

John McCarthy

Nathaniel Rochester

Allen Newell

Cliff Shaw

Herbert A. Simon

Oliver Selfridge

Frank Rosenblatt

Bernard Widrow

Joseph Weizenbaum

Seymour Papert

Seppo Linnainmaa

Paul Werbos

Geoffrey Hinton

John Hopfield

Jürgen Schmidhuber

Yann LeCun

Yoshua Bengio

Lotfi A. Zadeh

Stephen Grossberg

Alex Graves

James Goodnight

Andrew Ng

Fei-Fei Li

Alex Krizhevsky

Ilya Sutskever

Oriol Vinyals

Quoc V. Le

Ian Goodfellow

Demis Hassabis

David Silver

Andrej Karpathy

Ashish Vaswani
Noam Shazeer
Aidan Gomez
John Schulman
Mustafa Suleyman
Jan Leike
Daniel Kokotajlo
François Chollet
Neural Turing machine
Differentiable neural computer
Transformer Vision transformer (ViT)
Vision transformer (ViT)
Recurrent neural network (RNN)
Long short-term memory (LSTM)
Gated recurrent unit (GRU)
Echo state network
Multilayer perceptron (MLP)
Convolutional neural network (CNN)
Residual neural network (RNN)
Highway network
Mamba
Autoencoder
Variational autoencoder (VAE)
Generative adversarial network (GAN)
Graph neural network (GNN)
Category