# Continuous Space Word Vectors Obtained by Applying Word2Vec to Abstracts of Biomedical Articles

Ioannis Pavlopoulos[1], Aris Kosmopoulos[1,2] and Ion Androutsopoulos[1]

[1]NLP Group, Department of Informatics, Athens University of Economics and Business, Greece
http://nlp.cs.aueb.gr/

[2]Institute of Informatics and Telecommunications, NCSR "Demokritos", Greece
http://iit.demokritos.gr/

March 22, 2014

## Introduction

The *word2vec* tool[1] processes a large text corpus and maps the words of the corpus to vectors of a continuous space [1, 2, 3]. The word vectors can then be used, for example, to estimate the relatedness of two words or to perform query expansion.[2] During the European project BioASQ[3], we applied word2vec to 10,876,004 English abstracts[4] of biomedical articles from PubMed[5]. The resulting vectors of 1,701,632 distinct words (types) are now publicly available.

## How the word vectors were created

We first created a plain text file containing one abstract per line. We then used the *toolkit.py* script (see below) to create a one-line plain text file, containing all the abstracts (one after the other), with punctuation symbols, brackets etc. removed, all words separated by spaces, and converted to lower case. There were 9,395,602 distinct words in the resulting file.

We then used the *train_vectors.sh* script (see below) to apply word2vec (with the dimensionality of the vector space set to 200 and default values for other parameters) to the one-line text file with the abstracts. The resulting *.bin* file contains the word vectors produced by word2vec in binary format. The .bin file can be processed, for example, using *Gensim*[6]. Alternatively, the *toolkit.py* script can be used to load the .bin file and print its word vectors to a plain text file. We provide only the plain text file with the word vectors that our toolkit.py script produced, not the (much larger) .bin file, but the latter can also be generated using our scripts. By default, word2vec produces word vectors only for words that occur at least 5 times in the corpus (the abstracts). In our case, this led to vectors for 1,701,632 distinct words.

---

[1] Available from https://code.google.com/p/word2vec/.
[2] See http://tech.radialpoint.com/2014/03/11/word2vec-query-expansion-component-for-apache-lucene/.
[3] See http://www.bioasq.org/. The research leading to these results has received funding from the European Commission's 7th Framework Programme (FP7/2007-2013, ICT-2011.4.4(d), Intelligent Information Management, Targeted Competition Framework) under grant agreement n° 318652.
[4] The corpus is the training dataset of BioASQ Task 1a; see http://bioasq.lip6.fr/general_information/Task2a/.
[5] See http://www.ncbi.nlm.nih.gov/pubmed.
[6] See http://radimrehurek.com/gensim/models/word2vec.html.

## More information about the files provided

**toolkit.py:** A Python script that reads a plain text file containing one abstract per line. The script outputs a one-line plain text file containing all the abstracts (one after the other), with punctuation symbols, brackets etc. removed, all words separated by spaces, and converted to lower case. The resulting one-line file can then be used by train_vectors.sh. The toolkit.py script can also be used to read a .bin file produced by train_vectors.sh and a plain text file with a single word per line; let us call the second file the *words file*. In this case, toolkit.py prints the vectors (from the .bin file) of the words listed in the words file (in the same order) to a new plain text file, one vector per line. For words (in the words file) that there is no vector (in the .bin file), a line containing only "-1" is printed in the new text file.

**train_vectors.sh:** A shell script that reads a one-line plain text file with preprocessed abstracts (as produced by toolkit.py) and applies word2vec to the abstracts to create a .bin file with word vectors. The dimensionality of the vector space is set to 200. Default values are used for all the other parameters of word2vec. The one-line file with the abstracts and the executable code of word2vec must be in the same folder as train_vectors.sh.[7] In non-Unix operating systems, train_vectors.sh may have to be modified slightly.

**types.txt** and **vectors.txt**: These plain text files contain 1,701,632 lines each. The first file (types.txt) contains the distinct words (types) that we provide word vectors for, with a single word per line. The second file (vectors.txt) contains the word vectors of the corresponding words of the first file, one vector per line, each vector printed as 200 space-separated numbers.

## Examples of close words

As a quick demonstration of the possible uses of the word vectors, we collected the 100 most frequent words (excluding stop-words) of the 310 English biomedical questions of the development dataset of BioASQ Task 2b.[8] For each one of the 100 words, we found its three closest words (among the 1,701,632 distinct words we had vectors for), using the cosine similarity of the corresponding word vectors to measure the proximity between two words.[9] We then showed the following list, which shows the 100 words and their closest words, to two members of the BioASQ team of biomedical experts, who were involved in the construction of the benchmark datasets of Task 2b.[10] For each one of the 100 words, the two experts were asked to color the closest words as relevant (green), possibly relevant (orange), or irrelevant (red). The experts also marked any lines they found to contain spelling errors; these lines are shown in bold. It is difficult to define exactly when two words should be considered relevant, possibly relevant,

[7] Consult https://code.google.com/p/word2vec/ for information about compiling word2vec.
[8] See http://bioasq.lip6.fr/general_information/Task2b/.
[9] All of the 100 words also happened to be among the 1,701,632 words we had vectors for. We computed the cosine similarities using Gensim, which does not currently compute Euclidean distances.
[10] We are grateful to the two biomedical experts, whose names cannot be revealed.

or irrelevant, but we asked the experts to do their best. Most of the closest words were judged to be relevant (green).

protein: proteins, a-anchoring, pka-anchoring
thyroid: thyroidal, nonthyroid, hyperfunctioning
associated: correlated, related, correlates
**hormone: gh, luetinizing, fshluteinizing**
human: murine, mouse, immortalized
used: utilized, employed, applied
genes: gene, paralogs, operons
treatment: therapy, treatments, treating
**disease: diseases, disease-like, mmrn1rs6532197**
gene: genes, pseudogene, gene-encoding
heart: cardiac, chf, congestive
role: roles, plays, play
affect: alter, modify, impair
dna: dnas, bisulfite-treated, polymerase-mediated
histone: histones, h4k16, h4
involved: implicated, participates, regulating
list: lists, listing, to-do
proteins: protein, polypeptides, hsp70s
known: yet, presently, well-known
patients: outpatients, subjects, whom
present: this, aimed, our
**cancer: cancers, crc, caner**
receptor: receptors, hmc5, 5-nonyloxytryptamine
regulate: modulate, regulates, orchestrate
cell: cells, cancer-cell, sw1710
coding: 5-noncoding, 5-untranslated, 3-noncoding
inhibitors: inhibitor, small-molecule, atp-competing
many: several, some, numerous
related: linked, associated, relate
cardiomyopathy: cardiomyopathies, myocardiopathy, dcm
cause: causes, causing, sequela
children: adolescents, adults, toddlers
clinical: paraclinical, laboratorial, radiologic
common: frequent, prevalent, commonest
depression: anxiety, somatization, inventory-bdi
**drug: drugs, illicitstreet, analgesicantipyretic**
effect: effects, influence, impact
expression: over-expression, mrna, upregulation
mechanism: mechanisms, underlying, regulation
mutations: mutation, truncating, protein-truncating
name: names, fontibacillus, fulvivirga
thyroiditis: hashimotos, thyrotoxicosis, hyperthyroidism
tools: tool, methodologies, technologies
use: usage, users, misuse
**available: unavailable, httpcdbicsuciedu, avimaayanmssmedu**
brain: cerebrum, brains, cerebellum
drugs: agents, drug, medications
function: functions, dysfunction, impairment
genome: genomes, genomic, full-genome
mammaprint: 70-gene, 76-gene, 50-gene
multiple: stepwise, step-wise, polychotomous
**physical: leisure-time, leisure, activitysedentary**
receptors: receptor, agonists, gq11-coupled
reported: described, documented, claimed
**response: responses, responsiveness, reponse**
**signaling: signalling, signal-transduction, raserk**
splicing: pre-mrna, pre-mrnas, polyadenylation

test: tests, brunner-munzel, fisher-freeman-halton
trials: trial, rcts, well-designed
**activity: activities, acitivity, activites**
**blood: cellrbc, bloods, gasesph**
calcium: magnesium, ca2, potassium
diabetic: non-diabetic, nondiabetic, insulin-treated
diseases: pathologies, ailments, disease
disorder: disorders, hypocondriasis, mdd
effects: effect, actions, impact
found: observed, noted, noticed
insulin: c-peptide, hyperinsulinemia, glucagon
major: minor, main, principal
may: might, could, can
methyltransferases: methyltransferase, mtases, dnmt2
play: plays, played, pivotal
possible: causal, impossible, plausible
prediction: predicting, predictions, predication
proteomics: proteomic, metabolomics, lipidomics
regulated: orchestrated, regulates, regulate
**sequence: sequences, deduced, istaqtz2**
**sequences: sequence, exon-flanking, istaqtz2**
species: taxa, genera, morphospecies
subacute: sub-acute, acute, polyradiculoneuritis
syndrome: syndrome-like, syndrom, syndromes
**tissues: tissue, organs, tissuesorgans**
treat: manage, palliate, diagnose
triiodothyronine: tri-iodothyronine, 353-triiodothyronine, thyroxine
type: parenti-fraccaro, type-ii, i-trimer
**absorption: absorptions, absorbtion, flame-atomic**
acute: chronic, sub-acute, subacute
alpha1: alpha2, alpha3, beta2
analyses: analysis, bivariate, logisticlinear
analysis: analyses, analysis-based, logisticlinear
arthritis: jra, polyarthritis, ra
autism: autistic, asd, asds
babies: newborns, infants, neonates
best: good, reasonable, fit
beta1: β1, beta3, alphav
cancers: cancer, adenocarcinomas, carcinomas
carcinogenesis: tumorigenesis, tumourigenesis, hepatocarcinogenesis
cardiac: noncardiac, anesthesia-attributable, heart
chronic: acute, myeloleucosis, non-chronic
complete: near-complete, partial, incomplete

## References


[1] T. Mikolov, K. Chen, G. Corrado, and J. Dean. “Efficient Estimation of Word Representations in Vector Space”. In *Proceedings of Workshop at ICLR*, 2013.

[2] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. “Distributed Representations of Words and Phrases and their Compositionality”. In *Proceedings of NIPS*, 2013.

[3] T. Mikolov, W. Yih, and G. Zweig. “Linguistic Regularities in Continuous Space Word Representations”. In *Proceedings of NAACL HLT*, 2013.