

word2vec Demo

Efficient Estimation of Word Representations in Vector Space (Mikolov et al, 2013).

Andres Calderon and Hinna Shabir

University of California, Riverside

November 19, 2015

Agenda

1 Installation

2 Demos

- Word vectors
- Word classification
- Word analogies
- From words to phrases
- Word and phrase accuracy
- Pre-trained models
- Other implementations

Agenda

1 Installation

2 Demos

- Word vectors
- Word classification
- Word analogies
- From words to phrases
- Word and phrase accuracy
- Pre-trained models
- Other implementations

word2vec source code

- <https://code.google.com/p/word2vec/>.
- Provides an efficient implementation of the continuous bag-of-words and skip-gram.
- Clean and well documented code in C.

word2vec source code

```

and@and-dblab: /opt/word2vec
and@and-dblab: /opt/word2vec 112x24
and@and-dblab: /opt/word2vec$ svn checkout http://word2vec.googlecode.com/svn/trunk/
A trunk/word2phrase.c
A trunk/LICENSE
A trunk/word-analogy.c
A trunk/compute-accuracy.c
A trunk/demo-analogy.sh
A trunk/demo-classes.sh
A trunk/demo-train-big-model-v1.sh
A trunk/demo-word-accuracy.sh
A trunk/demo-phrases.sh
A trunk/questions-words.txt
A trunk/demo-phrase-accuracy.sh
A trunk/word.c
A trunk/distance.c
A trunk/README.txt
A trunk/questions-phrases.txt
A trunk/word2vec.c
A trunk/makefile
Checked out revision 42.
and@and-dblab: /opt/word2vec$

```

Labels
 NeuralNetwork, MachineLearning,
 NaturalLanguageProcessing,
 WordVectors, Google

Introduction

This tool provides an efficient implementation of the continuous bag-of-words and skip-gram architecture of words. These representations can be subsequently used in many natural language processing applications.

Quick start

- Download the code: svn checkout <http://word2vec.googlecode.com/svn/trunk/>
- Run 'make' to compile word2vec tool

word2vec source code

```

and@and-dblab: /opt/word2vec/trunk
and@and-dblab: /opt/word2vec/trunk 112x24
and@and-dblab: /opt/word2vec/trunk$ more makefile
CC = gcc
#Using -Ofast instead of -O3 might result in faster code, but is supported only by newer GCC versions
CFLAGS = -lm -pthread -O3 -march=native -Wall -funroll-loops -Wno-unused-result

all: word2vec word2phrase distance word-analogy compute-accuracy
word2vec: word2vec.c
$(CC) word2vec.c -o word2vec $(CFLAGS)
word2phrase: word2phrase.c
$(CC) word2phrase.c -o word2phrase $(CFLAGS)
distance: distance.c
$(CC) distance.c -o distance $(CFLAGS)
word-analogy: word-analogy.c
$(CC) word-analogy.c -o word-analogy $(CFLAGS)
compute-accuracy: compute-accuracy.c
$(CC) compute-accuracy.c -o compute-accuracy $(CFLAGS)
chmod +x *.sh
Code license
Apache License 2.0
clean: rm -rf word2vec word2phrase distance word-analogy compute-accuracy
and@and-dblab: /opt/word2vec/trunk$

```

Project Summary

READ-ONLY: This project has been archived. For more information see [this post](#).

Introduction

This tool provides an efficient implementation of the continuous bag-of-words and skip-gram architecture of words. These representations can be subsequently used in many natural language processing applications.

Quick start

- Download the code: svn checkout <http://word2vec.googlecode.com/svn/trunk/>
- Run 'make' to compile word2vec tool

word2vec source code

```

and@and-dblab: /opt/word2vec/trunk
and@and-dblab: /opt/word2vec/trunk 112x28
and@and-dblab: /opt/word2vec/trunk$ ls -la
total 1036
drwxrwxr-x 3 and and 4096 Nov 18 10:59 P Web
drwxrwxr-x 3 and and 4096 Nov 18 10:52 ..
-rwxrwxr-x 1 and and 13672 Nov 18 10:59 compute-accuracy
-rw-rw-r-- 1 and and 5241 Nov 18 10:52 compute-accuracy.c
-rwxrwxr-x 1 and and 631 Nov 18 10:52 demo-analogy.sh
-rwxrwxr-x 1 and and 358 Nov 18 10:52 demo-classes.sh
-rwxrwxr-x 1 and and 885 Nov 18 10:52 demo-phrase-accuracy.sh
-rwxrwxr-x 1 and and 853 Nov 18 10:52 demo-phrases.sh
-rwxrwxr-x 1 and and 5126 Nov 18 10:52 demo-train-big-model-v1.sh
-rwxrwxr-x 1 and and 414 Nov 18 10:52 demo-word-accuracy.sh
-rwxrwxr-x 1 and and 272 Nov 18 10:52 demo-word.sh
-rwxrwxr-x 1 and and 21736 Nov 18 10:59 distance
-rw-rw-r-- 1 and and 4557 Nov 18 10:52 distance.c
-rw-rw-r-- 1 and and 11358 Nov 18 10:52 instructions
-rw-rw-r-- 1 and and 718 Nov 18 10:52 makefile
-rw-rw-r-- 1 and and 168209 Nov 18 10:52 questions-phrases.txt
-rw-rw-r-- 1 and and 603955 Nov 18 10:52 questions-words.txt
-rw-rw-r-- 1 and and 1209 Nov 18 10:52 README.txt
drwxrwxr-x 4 and and 4096 Nov 18 10:52 svn
-rwxrwxr-x 1 and and 23016 Nov 18 10:59 word2phrase
-rw-rw-r-- 1 and and 9386 Nov 18 10:52 word2phrase.c
-rwxrwxr-x 1 and and 57416 Nov 18 10:59 word2vec
-rw-rw-r-- 1 and and 26184 Nov 18 10:52 word2vec.c
-rwxrwxr-x 1 and and 21688 Nov 18 10:59 word-analogy
-rw-rw-r-- 1 and and 4664 Nov 18 10:52 word-analogy.c
and@and-dblab: /opt/word2vec/trunk$

```

word2vec source code



and@and-dblab: /opt/word2vec/trunk
and@and-dblab: /opt/word2vec/trunk 131x46

and@and-dblab: /opt/word2vec/trunk word2vec/
WORD VECTOR estimation toolkit v 0.1.0

Options:
Parameters for training:
-train <file>
Use <file> to train the model
-output <file>
Use <file> to save the resulting word vectors / word clusters
-size <int>
Set size of word vectors; default is 100
-window <int>
Set max skip length between words; default is 5
-sample <float>
Set threshold for occurrence of words. Those that appear with higher frequency in the training data will be randomly down-sampled. Default value is 1e-3, useful range is (0, 1e-5)
-negative <int>
Use Hierarchical Softmax; default is 0 (not used)
-ns <int>
Number of negative examples; default is 5, common values are 3 - 10 (0 - not used)
-threads <int>
Use <int> threads (default: 1)
-min-count <int>
This will discard words that appear less than <int> times; default is 5
-alpha <float>
Set the starting learning rate; default is 0.025 for skip-gram and 0.05 for CBOW
-classes <int>
Output word classes rather than word vectors; default number of classes is 0 (vectors are written)
-debug <int>
The debug mode (default is 0, not used) produces the word vectors as output. It first constructs a vocabulary from the input corpus and then uses it to produce the word vectors. The resulting word vector file can be used as features in many natural language processing applications.
-save-vocab <file>
Save the resulting vectors in binary mode; default is 0 (off)
-read-vocab <file>
The vocabulary will be read from <file>, not constructed from the training data
-cbow <int>
Use the continuous bag of words model; default is 1 (use 0 for skip-gram model)
Examples:
./word2vec -train data.txt -output vec.txt -size 200 -window 5 -sample 1e-4 -negative 5 -hs 0 -binary 0 -cbow 1 -iter 3
and@and-dblab: /opt/word2vec/trunk\$

Word Correlation distance

spain	0.678515
netherlands	0.652428
italy	0.633238
switzerland	0.622323
portugal	0.577154
ruusia	0.571587
germany	0.563291
france	0.543146

Agenda

1 Installation

2 Demos

- Word vectors
- Word classification
- Word analogies
- From words to phrases
- Word and phrase accuracy
- Pre-trained models
- Other implementations

Agenda

1 Installation

2 Demos

- Word vectors
- Word classification
- Word analogies
- From words to phrases
- Word and phrase accuracy
- Pre-trained models
- Other implementations

demo-word.sh

```
## Compile the code...
make
## Download and unzip the training file...
if [ ! -e text8 ]; then
    wget http://mattmahoney.net/dc/text8.zip -O text8.gz
    gzip -d text8.gz -f
fi
## Run the model (taking time)...
time ./word2vec -train text8 -output vectors.bin -cbow 1 -size 200 -window 8 -negative 25 -hs 0
    ↪ -sample 1e-4 -threads 20 -binary 1 -iter 15
## Query word distances...
./distance vectors.bin
```

12 / 47

demo-word.sh output

```
and@and-dblab:~/Documents/Projects/C++/word2vec/trunk$ ./demo-word.sh
make: Nothing to be done for 'all'.
--2015-11-12 18:18:10-- http://mattmahoney.net/dc/text8.zip
Resolving mattmahoney.net (mattmahoney.net)... 98.139.135.129
Connecting to mattmahoney.net (mattmahoney.net)|98.139.135.129|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 31344016 (30M) [application/zip]
Saving to: text8.gz

text8.gz
  ↪ 100%[=====>] 29.89M 1.74MB/s in 18s

2015-11-12 18:18:28 (1.70 MB/s) - text8.gz saved [31344016/31344016]

Starting training using file text8
Vocab size: 71291
Words in train file: 16718843
Alpha: 0.000005 Progress: 100.10% Words/thread/sec: 113.47k
real 10m15.450s
user 36m52.552s
sys 0m4.388s
Enter word or sentence (EXIT to break):
```

demo-00.sh

```
## Get a small file...
head -c 5000000 text8 > text8_small
## Build the model...
./word2vec -train text8_small -output vectors_small.bin -cbow 1 -size 100 -window 5 -negative 0 -hs
↪ 25 -threads 1 -iter 4 -min-count 2 -binary 1
## Query word distances...
./distance vectors_small.bin
```

demo-01.sh

```
## Text model saving vocabulary
./word2vec -train text8_small -output vectors_small_50.txt -cbow 1 -size 50 -window 5 -negative 0
↳ -hs 25 -threads 1 -iter 4 -binary 0 -save-vocab vocab.txt
## Text model with just 3 dimensions
./word2vec -train text8_small -output vectors_small_3.txt -cbow 1 -size 3 -window 5 -negative 0 -hs
↳ 25 -threads 1 -iter 4 -binary 0
## See the results...
echo "Text model size 50..."
head -n 5 vectors_small_50.txt
echo "Vocabulary..."
head -n 5 vocab.txt
echo "Text model size 3..."
head -n 5 vectors_small_3.txt
```

demo-word.sh revisited

- **distance** can load a pre-trained model...
- Let's try some examples...
 - 1 california
 - 2 sciences
 - 3 happiness
 - 4 man
 - 5 ...

Agenda

1 Installation

2 Demos

- Word vectors
- **Word classification**
- Word analogies
- From words to phrases
- Word and phrase accuracy
- Pre-trained models
- Other implementations

demo-classes.sh

```
## Same than before...
make
if [ ! -e text8 ]; then
    wget http://mattmahoney.net/dc/text8.zip -O text8.gz
    gzip -d text8.gz -f
fi
## Train the model with classes rather than vectors...
time ./word2vec -train text8 -output classes.txt -cbow 1 -size 200 -window 8 -negative 25 -hs 0
    ↪ -sample 1e-4 -threads 20 -iter 15 -classes 500
## Sort the result by the second column...
sort classes.txt -k 2 -n > classes.sorted.txt
echo The word classes were saved to file classes.sorted.txt
```

demo-classes.sh

```
## Let's build a small model...  
./word2vec -train text8_small -output classes.txt -cbow 1 -size 50 -window 5 -negative 0 -hs  
  ↪ 12 -sample 1e-4 -threads 20 -iter 3 -classes 10  
sort classes.txt -k 2 -n > classes.sorted.txt
```

Agenda

1 Installation

2 Demos

- Word vectors
- Word classification
- **Word analogies**
- From words to phrases
- Word and phrase accuracy
- Pre-trained models
- Other implementations

Interesting properties of the word vectors

- $\overrightarrow{\text{paris}} - \overrightarrow{\text{france}} + \overrightarrow{\text{italy}} \cong \overrightarrow{\text{rome}}$
- $\overrightarrow{\text{king}} - \overrightarrow{\text{man}} + \overrightarrow{\text{woman}} \cong \overrightarrow{\text{queen}}$

demo-analogy.sh

```
## Same that before...
make
if [ ! -e text8 ]; then
    wget http://mattmahoney.net/dc/text8.zip -O text8.gz
    gzip -d text8.gz -f
fi
echo -----
echo Note that for the word analogy to perform well, the model
echo should be trained on much larger data set
echo Example input: paris france berlin
echo -----
time ./word2vec -train text8 -output vectors.bin -cbow 1 -size 200 -window 8 -negative 25 -hs 0
    ↪ -sample 1e-4 -threads 20 -binary 1 -iter 15
## Call word-analogy script...
./word-analogy vectors.bin
```

demo-analogy.sh

- Some examples...

- 1 paris france bogota ...
- 2 king man queen ...
- 3 boy girl brother ...
- 4 chicago illinois memphis ...
- 5 poland zloty sweden ...
- 6 bad worst good ...
- 7 child children mouse ...
- 8 going went selling ...
- 9 mexico mexican peru ...
- 10 berlin germany riyadh¹ ...
- 11 woman angel man ...
- 12 heaven hell man ...

¹word2phrase will address the problem...

Agenda

1 Installation

2 Demos

- Word vectors
- Word classification
- Word analogies
- **From words to phrases**
- Word and phrase accuracy
- Pre-trained models
- Other implementations

From words to phrases and beyond

- It is desirable to have only one vector for representing 'san_francisco'.
- How to get vector representation of larger pieces of text no just words?
- **word2phrase**
- Pre-processing the training data set to form phrases.

demo-phrases.sh

```

and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk 120x22
and@and-dblab:~/Documents/Projects/C++/word2vec/trunk$ ./word2phrase
WORD2PHRASE tool v0.1a
Options:
Parameters for training:
  -train <file>          Use text data from <file> to train the model
  -output <file>         Use <file> to save the resulting word vectors / word clusters / phrases
  -min-count <int>       This will discard words that appear less than <int> times; default is 5
  -threshold <float>     <float> value represents threshold for forming the phrases (higher means less phrases); default 100
  -debug <int>           Set the debug mode (default 0 = more info during training)

Examples:
./word2phrase -train text.txt -output phrases.txt -threshold 100 -debug 2

and@and-dblab:~/Documents/Projects/C++/word2vec/trunk$
and@and-dblab:~/Documents/Projects/C++/word2vec/trunk$
and@and-dblab:~/Documents/Projects/C++/word2vec/trunk$

```

There are two main learning algorithms in word2vec : continuous bag-of-words and continuous skip-gram. The skip-gram model is more robust to noise and is able to pick one of these learning algorithms. Both algorithms learn the representation of a word that is useful for predicting context. These models are described in detail in [1,2].

Interesting properties of the word vectors

There are strong regularities in the word vector space, for example vector operations $vector(Italy) - vector(Rome)$ results in a vector that is very close to $vector(king) - vector(man) + vector(woman)$. This can be achieved by pre-processing the training data set to form the phrases used in the word2vec script `/demo-phrases.sh`. The example output with the closest tokens to 'san_francisco' is

From words to phrases and beyond

demo-phrases.sh

```

## Compile...
make
## Download...
if [ ! -e news.2012.en.shuffled ]; then
    wget http://www.statmt.org/wmt14/training-monolingual-news-crawl/news.2012.en.shuffled.gz
    gzip -d news.2012.en.shuffled.gz
fi
## Pre-process...
sed -e "s/'/'/g" -e "s/'/'/g" -e "s/'/'/g" < news.2012.en.shuffled | tr -c "A-Za-z'_ \n" " " >
    ↪ news.2012.en.shuffled-norm0
time ./word2phrase -train news.2012.en.shuffled-norm0 -output news.2012.en.shuffled-norm0-phrase0
    ↪ -threshold 200 -debug 2
time ./word2phrase -train news.2012.en.shuffled-norm0-phrase0 -output
    ↪ news.2012.en.shuffled-norm0-phrase1 -threshold 100 -debug 2
tr A-Z a-z < news.2012.en.shuffled-norm0-phrase1 > news.2012.en.shuffled-norm1-phrase1
## Model...
time ./word2vec -train news.2012.en.shuffled-norm1-phrase1 -output vectors-phrase.bin -cbow 1 -size
    ↪ 200 -window 10 -negative 25 -hs 0 -sample 1e-5 -threads 20 -binary 1 -iter 15
## Deploy...
./distance vectors-phrase.bin

```

demo-phrases.sh

```

and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk 120x22
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk$ head -n 12 news.2012.en.shuffled-norm0
Prang's initial success came from small prints and cards including the first Christmas cards but with the arrival of
the war he began to issue maps.
This single mining project is one of the main reasons for the amazing economic growth in the country said Dale Ch
oi an analyst at Origo Partners a private equity company that advises investors on China and Mongolia
On Dec six days before Christmas Jocelyn Earnest's close friend Marcy Shepherd who had been texting with h
er all day became concerned when Jocelyn never responded to messages she sent that evening
It is a phenomenon affecting the whole industry
British male solo artist Ed Sheeran
On Sunday morning the clean up crew jumped at the sight of a scurrying rodent shouting rat as it scrambled under t
he roof.
If you accidentally slide past the base you don't get called out of the baseline he said adding that qualified in th
is case
Damage to both ships is being evaluated with both ships currently operating under their own power
Next year it will also replace its Douro River ship in Portugal which stops in Porto and the wine growing town of Pinh
o with the motor passenger Queen Isabel
He tried to cross the street
The Labour MP Clive Betts will captain the other side
They are very stable they have stuck with the same starting lineup and don't expect many changes except for the injured
striker Helder Postiga
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk$
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk$

```

demo-phrases.sh

```

and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk 120x22
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk$ head -n 12 news.2012.en.shuffled-norm0-phrase0
Prang's initial success came from small prints and cards including the first Christmas cards but with the arrival of th
e war he began to issue maps Calendar R Web Google Translate Ph.D. Misc Riverside Apartm Riverside Studen
This single mining project is one of the main reasons for the amazing economic growth in the country said Dale Choi an
analyst at Origo Partners a private equity company that advises investors on China and Mongolia
On Dec six days before Christmas Jocelyn Earnest's close friend Marcy Shepherd who had been texting with her all day be
came concerned when Jocelyn never responded to messages she sent that evening
It is a phenomenon affecting the whole industry
British male solo artist Ed Sheeran
On Sunday morning the clean up crew jumped at the sight of a scurrying rodent shouting rat as it scrambled under their
feet
If you accidentally slide past the base you don't get called out of the baseline he said adding that qualified in this
case
Summary People
Damage to both ships is being evaluated with both ships currently operating under their own power
Next year it will also replace its Douro River ship in Portugal which stops in Porto and the wine growing town of Pinh
o with the passenger Queen Isabel
He tried to cross the street
The Labour MP Clive Betts will captain the other side
They are very stable they have stuck with the same starting lineup and don't except any change except for the injured s
triker Helder Postiga
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk$
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk$

```

demo-phrases.sh

```

and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk 120x22
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk$ head -n 12 news.2012.en.shuffled-norm0-phrases1
Prang's initial success came from small prints and cards including the first Christmas cards but with the arrival of th
e war he began to issue maps Calendar R Web Google Translate Ph.D. Misc Riverside Apartm Riverside Studen
This single mining project is one of the main reasons for the amazing economic growth in the country said Dale Choi an
analyst at Origo Partners a private equity company that advises investors on China and Mongolia
On Dec six days before Christmas Jocelyn Earnest's close friend Marcy Shepherd who had been texting with her all day be
came concerned when Jocelyn never responded to messages she sent that evening
It is a phenomenon affecting the whole industry
British male solo artist's Sheeran
On Sunday morning the clean up crew jumped at the sight of a scurrying rodent shouting rat as it scrambled under their
feet
If you accidentally slide past the base you don't get called out of the baseline he said adding that qualified in this
case
Damage to both ships is being evaluated with both ships currently operating under their own power
Next year it will also replace its Douro River ship in Portugal which stops in Porto and the wine growing town of Pinh
o with the passenger Queen Isabel
He tried to cross the street
The Labour MP Clive Betts will captain the other side
They are very stable they have stuck with the same starting lineup and don't except any change except for the injured s
triker Helder Postiga
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk$
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk$

```

demo-phrases.sh output (1/3)

```
and@and-dblab:~/Documents/Projects/C++/word2vec/trunk$ ./demo-phrases.sh
make: Nothing to be done for 'all'.
--2015-11-12 18:33:08--
  ↪ http://www.statmt.org/wmt14/training-monolingual-news-crawl/news.2012.en.shuffled.gz
Resolving www.statmt.org (www.statmt.org)... 129.215.197.100
Connecting to www.statmt.org (www.statmt.org)|129.215.197.100|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 786717767 (750M) [application/x-gzip]
Saving to: news.2012.en.shuffled.gz

news.2012.en.shuffled.gz
  ↪ 100%[=====>] 750.27M  4.25MB/s   in 3m 7s

2015-11-12 18:36:16 (4.02 MB/s) - news.2012.en.shuffled.gz saved [786717767/786717767]

...
```

demo-phrases.sh output (2/3)

```
...
Starting training using file news.2012.en.shuffled-norm0
Words processed: 296900K      Vocab size: 33198K
Vocab size (unigrams + bigrams): 18838711
Words in train file: 296901342
Words written: 296900K
real  7m38.607s
user  7m8.592s
sys   0m15.176s

Starting training using file news.2012.en.shuffled-norm0-phrase0
Words processed: 280500K      Vocab size: 38761K
Vocab size (unigrams + bigrams): 21728781
Words in train file: 280513979
Words written: 280500K
real  7m0.022s
user  6m19.436s
sys   0m14.756s
...
```


demo-phrases.sh output (3/3)

```
...
Starting training using file news.2012.en.shuffled-norm1-phrase1
Vocab size: 681320
Words in train file: 283545447
Alpha: 0.000005 Progress: 100.00% Words/thread/sec: 162.97k
real 115m6.531s
user 434m57.904s
sys 1m4.464s
Enter word or sentence (EXIT to break):
```

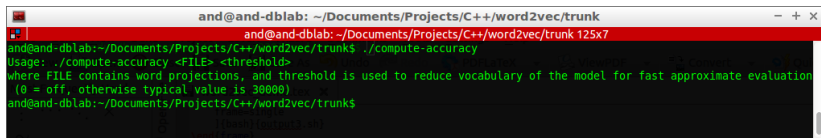
Agenda

1 Installation

2 Demos

- Word vectors
- Word classification
- Word analogies
- From words to phrases
- **Word and phrase accuracy**
- Pre-trained models
- Other implementations

compute-accuracy



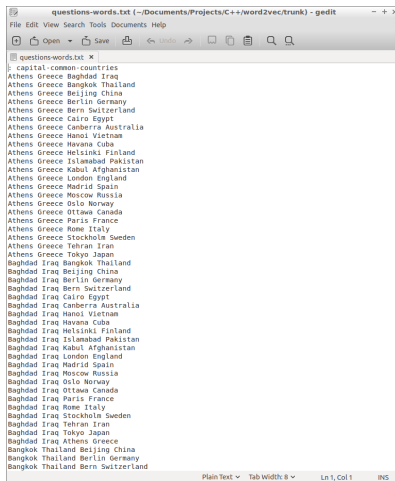
```
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk 125x7
and@and-dblab:~/Documents/Projects/C++/word2vec/trunk$ ./compute-accuracy
Usage: ./compute-accuracy <FILE> <threshold>
where FILE contains word projections, and threshold is used to reduce vocabulary of the model for fast approximate evaluation
(0 = off, otherwise typical value is 30000)
and@and-dblab:~/Documents/Projects/C++/word2vec/trunk$
```

demo-word-accuracy.sh

```
## Same than before...
make
if [ ! -e text8 ]; then
    wget http://mattmahoney.net/dc/text8.zip -O text8.gz
    gzip -d text8.gz -f
fi
time ./word2vec -train text8 -output vectors.bin -cbow 1 -size 200 -window 8 -negative 25 -hs 0
    ↪ -sample 1e-4 -threads 20 -binary 1 -iter 15
## Test accuracy...
./compute-accuracy vectors.bin 30000 < questions-words.txt
## to compute accuracy with the full vocabulary, use: ./compute-accuracy vectors.bin <
    ↪ questions-words.txt
```

questions-words.txt

• <https://word2vec.googlecode.com/svn/trunk/questions-words.txt>



```

questions-words.txt (-/Documents/Projects/C++/word2vec/trunk) - gedit
File Edit View Search Tools Documents Help
Open Save Undo Redo Find
questions-words.txt x
capital-common countries
Athens Greece Baghdad Iraq
Athens Greece Bangkok Thailand
Athens Greece Beijing China
Athens Greece Berlin Germany
Athens Greece Bern Switzerland
Athens Greece Cairo Egypt
Athens Greece Canberra Australia
Athens Greece Hanoi Vietnam
Athens Greece Havana Cuba
Athens Greece Helsinki Finland
Athens Greece Islamabad Pakistan
Athens Greece Kabul Afghanistan
Athens Greece London England
Athens Greece Madrid Spain
Athens Greece Moscow Russia
Athens Greece Oslo Norway
Athens Greece Ottawa Canada
Athens Greece Paris France
Athens Greece Rome Italy
Athens Greece Stockholm Sweden
Athens Greece Tehran Iran
Athens Greece Tokyo Japan
Baghdad Iraq Bangkok Thailand
Baghdad Iraq Beijing China
Baghdad Iraq Berlin Germany
Baghdad Iraq Bern Switzerland
Baghdad Iraq Cairo Egypt
Baghdad Iraq Canberra Australia
Baghdad Iraq Hanoi Vietnam
Baghdad Iraq Havana Cuba
Baghdad Iraq Helsinki Finland
Baghdad Iraq Islamabad Pakistan
Baghdad Iraq Kabul Afghanistan
Baghdad Iraq London England
Baghdad Iraq Madrid Spain
Baghdad Iraq Moscow Russia
Baghdad Iraq Oslo Norway
Baghdad Iraq Ottawa Canada
Baghdad Iraq Paris France
Baghdad Iraq Rome Italy
Baghdad Iraq Stockholm Sweden
Baghdad Iraq Tehran Iran
Baghdad Iraq Tokyo Japan
Baghdad Iraq Athens Greece
Bangkok Thailand Beijing China
Bangkok Thailand Berlin Germany
Bangkok Thailand Bern Switzerland

```

demo-word-accuracy.sh output

```

and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk$ ./demo-word-accuracy.sh
capital-common-countries:
ACCURACY TOP1: 83.00 % (420 / 506)
Total accuracy: 83.00 % Semantic accuracy: 83.00 % Syntactic accuracy: -nan %
capital-world:
ACCURACY TOP1: 62.67 % (910 / 1452)
Total accuracy: 67.93 % Semantic accuracy: 67.93 % Syntactic accuracy: -nan %
currency:
ACCURACY TOP1: 20.90 % (56 / 268)
Total accuracy: 62.26 % Semantic accuracy: 62.26 % Syntactic accuracy: -nan %
city-in-state:
ACCURACY TOP1: 49.01 % (770 / 1571)
Total accuracy: 56.78 % Semantic accuracy: 56.78 % Syntactic accuracy: -nan %
family:
ACCURACY TOP1: 77.78 % (238 / 306)
Total accuracy: 58.35 % Semantic accuracy: 58.35 % Syntactic accuracy: -nan %
gram1-adjective-to-adverb:
ACCURACY TOP1: 18.25 % (138 / 756)
Total accuracy: 52.11 % Semantic accuracy: 58.35 % Syntactic accuracy: 18.25 %
gram2-opposite:
ACCURACY TOP1: 23.53 % (72 / 306)
Total accuracy: 50.42 % Semantic accuracy: 58.35 % Syntactic accuracy: 19.77 %
gram3-comparative:
ACCURACY TOP1: 62.22 % (784 / 1260)
Total accuracy: 52.73 % Semantic accuracy: 58.35 % Syntactic accuracy: 42.81 %
gram4-superlative:
ACCURACY TOP1: 38.93 % (197 / 506)
Total accuracy: 51.72 % Semantic accuracy: 58.35 % Syntactic accuracy: 42.11 %
gram5-present-participle:
ACCURACY TOP1: 39.31 % (390 / 992)
Total accuracy: 50.17 % Semantic accuracy: 58.35 % Syntactic accuracy: 41.39 %
gram6-nationality-adjective:
ACCURACY TOP1: 86.29 % (1183 / 1371)
Total accuracy: 55.50 % Semantic accuracy: 58.35 % Syntactic accuracy: 53.25 %
gram7-past-tense:
ACCURACY TOP1: 38.21 % (509 / 1332)
Total accuracy: 53.33 % Semantic accuracy: 58.35 % Syntactic accuracy: 50.18 %
gram8-plural:
ACCURACY TOP1: 63.71 % (632 / 992)
Total accuracy: 54.22 % Semantic accuracy: 58.35 % Syntactic accuracy: 51.96 %
gram9-plural-verbs:
ACCURACY TOP1: 34.77 % (226 / 650)
Total accuracy: 53.19 % Semantic accuracy: 58.35 % Syntactic accuracy: 50.59 %
Questions seen / total: 12268 19544 62.77 %
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk$

```

demo-phrase-accuracy.sh

```
## Same than before...
make
if [ ! -e news.2012.en.shuffled ]; then
    wget http://www.statmt.org/wmt14/training-monolingual-news-crawl/news.2012.en.shuffled.gz
    gzip -d news.2012.en.shuffled.gz -f
fi
sed -e "s/'/'/g" -e "s/'/'/g" -e "s/'/'/ /g" < news.2012.en.shuffled | tr -c "A-Za-z'_ \n" " " >
    ↪ news.2012.en.shuffled-norm0
time ./word2phrase -train news.2012.en.shuffled-norm0 -output news.2012.en.shuffled-norm0-phrase0
    ↪ -threshold 200 -debug 2
time ./word2phrase -train news.2012.en.shuffled-norm0-phrase0 -output
    ↪ news.2012.en.shuffled-norm0-phrase1 -threshold 100 -debug 2
tr A-Z a-z < news.2012.en.shuffled-norm0-phrase1 > news.2012.en.shuffled-norm1-phrase1
time ./word2vec -train news.2012.en.shuffled-norm1-phrase1 -output vectors-phrase.bin -cbow 1 -size
    ↪ 200 -window 10 -negative 25 -hs 0 -sample 1e-5 -threads 20 -binary 1 -iter 15
## Test accuracy...
./compute-accuracy vectors-phrase.bin < questions-phrases.txt
```

questions-phrases.txt

https://word2vec.googlecode.com/svn/trunk/questions-phrases.txt

```

questions-phrases.txt (-/Documents/Projects/C++/word2vec/trunk) - gedit
File Edit View Search Tools Documents Help
Open Save Undo Redo Find
questions-phrases.txt *
:
newspapers
Albuquerque Albuquerque Journal Baltimore Baltimore Sun
Albuquerque Albuquerque Journal Boston Boston Globe
Albuquerque Albuquerque Journal Cincinnati Cincinnati Enquirer
Albuquerque Albuquerque Journal Cleveland Cleveland Plain Dealer
Albuquerque Albuquerque Journal Charleston Charleston Gazette
Albuquerque Albuquerque Journal Chicago Chicago Tribune
Albuquerque Albuquerque Journal Columbus Columbus Dispatch
Albuquerque Albuquerque Journal Dallas Dallas Morning News
Albuquerque Albuquerque Journal Dayton Dayton Daily News
Albuquerque Albuquerque Journal Denver Denver Post
Albuquerque Albuquerque Journal Dothan Dothan Eagle
Albuquerque Albuquerque Journal Fort Collins Fort Collins Coloradoan
Albuquerque Albuquerque Journal Fresno Fresno Bee
Albuquerque Albuquerque Journal Houston Houston Chronicle
Albuquerque Albuquerque Journal Indianapolis Indianapolis Star
Albuquerque Albuquerque Journal Knoxville Knoxville News Sentinel
Albuquerque Albuquerque Journal Los Angeles Los Angeles Times
Albuquerque Albuquerque Journal Miami Miami Herald
Albuquerque Albuquerque Journal Milwaukee Milwaukee Journal Sentinel
Albuquerque Albuquerque Journal Minneapolis Minneapolis Star Tribune
Albuquerque Albuquerque Journal New Haven New Haven Register
Albuquerque Albuquerque Journal New York New York Times
Albuquerque Albuquerque Journal Oakland Oakland Tribune
Albuquerque Albuquerque Journal Philadelphia Philadelphia Inquirer
Albuquerque Albuquerque Journal Portland Portland Oregonian
Albuquerque Albuquerque Journal Sacramento Sacramento Bee
Albuquerque Albuquerque Journal Salt Lake Salt Lake Tribune
Albuquerque Albuquerque Journal San Antonio San Antonio Express News
Albuquerque Albuquerque Journal San Francisco San Francisco Chronicle
Albuquerque Albuquerque Journal San Jose San Jose Mercury News
Albuquerque Albuquerque Journal Seattle Seattle Times
Albuquerque Albuquerque Journal Tallahassee Tallahassee Democrat
Albuquerque Albuquerque Journal Waco Waco Tribune Herald
Albuquerque Albuquerque Journal Washington Washington Post
Albuquerque Albuquerque Journal Worcester Worcester Telegram
Baltimore Baltimore Sun Boston Boston Globe
Baltimore Baltimore Sun Cincinnati Cincinnati Enquirer
Baltimore Baltimore Sun Cleveland Cleveland Plain Dealer
Baltimore Baltimore Sun Charleston Charleston Gazette
Baltimore Baltimore Sun Chicago Chicago Tribune
Baltimore Baltimore Sun Columbus Columbus Dispatch
Baltimore Baltimore Sun Dallas Dallas Morning News
Baltimore Baltimore Sun Dayton Dayton Daily News
Baltimore Baltimore Sun Denver Denver Post
Baltimore Baltimore Sun Dothan Dothan Eagle
Baltimore Baltimore Sun Fort Collins Fort Collins Coloradoan
Baltimore Baltimore Sun Fresno Fresno Bee
Plain Text Tab Width: 8 Ln 1, Col 1 INS

```


demo-phrase-accuracy.sh output

```

and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk
and@and-dblab: ~/Documents/Projects/C++/word2vec/trunk 86x21
and@and-dblab:~/Documents/Projects/C++/word2vec/trunk$ ./demo-phrase-accuracy.sh
newspapers:
ACCURACY TOP1: 27.84 % (294 / 1056)
Total accuracy: 27.84 % Semantic accuracy: 27.84 % Syntactic accuracy: -nan %
ice hockey:
ACCURACY TOP1: 17.52 % (123 / 702)
Total accuracy: 23.72 % Semantic accuracy: 23.72 % Syntactic accuracy: -nan %
basketball:
ACCURACY TOP1: 60.83 % (365 / 600)
Total accuracy: 33.16 % Semantic accuracy: 33.16 % Syntactic accuracy: -nan %
airlines:
ACCURACY TOP1: 23.81 % (100 / 420)
Total accuracy: 31.75 % Semantic accuracy: 31.75 % Syntactic accuracy: -nan %
people-companies:
ACCURACY TOP1: 82.29 % (79 / 96)
Total accuracy: 33.44 % Semantic accuracy: 33.44 % Syntactic accuracy: -nan %
Questions seen / total: 2874 3218 89.31 %
and@and-dblab:~/Documents/Projects/C++/word2vec/trunk$

```

Agenda

1 Installation

2 Demos

- Word vectors
- Word classification
- Word analogies
- From words to phrases
- Word and phrase accuracy
- **Pre-trained models**
- Other implementations

demo-train-big-model-v1.sh

```
#####  
#  
# Script for training good word and phrase vector model using public corpora, version 1.0.  
# The training time will be from several hours to about a day.  
#  
# Downloads about 8 billion words, makes phrases using two runs of word2phrase, trains  
# a 500-dimensional vector model and evaluates it on word and phrase analogy tasks.  
#  
#####
```

- GoogleNews-vectors-negative300.bin.gz
 - 100 billion words
 - 300 dimensional vectors
 - 1.6 GB
- freebase-vectors-skipgram1000.bin.gz
 - 100 billion words
 - 1000 dimensional vectors
 - 2.5 GB
- Some tips about performance and where obtain more training data can be found in <https://code.google.com/p/word2vec/>.

Agenda

1 Installation

2 Demos

- Word vectors
- Word classification
- Word analogies
- From words to phrases
- Word and phrase accuracy
- Pre-trained models
- Other implementations

Ports of the framework...

- in Python: gensim²
- in Java: deeplearning4j³
- in R: tmcn⁴
- in Scala: Apache Spark⁵

²<http://rare-technologies.com/deep-learning-with-word2vec-and-gensim/>

³<http://deeplearning4j.org/word2vec.html>

⁴<http://rpackages.ianhowson.com/rforge/tmcn.word2vec/man/word2vec.html>

⁵<https://spark.apache.org/docs/latest/mllib-feature-extraction.html#word2vec>

Thank you!!!

Download this presentation at <https://goo.gl/pE8Mjw>