Karlsruher Institut für Technologie

Karlsruhe Institute of Technology
Ngoc-Quan Pham
Institute for Anthropomatics and Robotics, Interactive Systems Labs
ngoc.pham@kit.edu

MT Praktikum SOLUTION- Word Embeddings & NNLM

3. Juli 2018

Environment Setup
First we need to have access to our cluster enviroment.
If you can use the available machines in the pool room, please log in with username: smt[30-45] (any number between 30 to 45, for example smt35), password=123456.
If you use your own laptop, you can directly connect to the cluster using ssh, using this command:
ssh smt[30-45]@i13hpc1.ira.uka.de

For today's work we are going to need better computing power. Please log into i13hpc28 or i13hpc29, using the following command (once you are in a terminal of the pool room's computers or after ssh from your own computer):
ssh i13hpc28 **or** ssh i13hpc29
From there, go to the working directory cd /project/smtstud/ss18/systems/

Now enter the virtual environment which is pre-installed using 2 commands:
bash
. /project/smtstud/ss18/commands/setup.sh
Finally, create a working directory with
`mkdir -p /project/smtstud/ss18/systems/USERNAME/`
(USERNAME is anything you like (smtxx or your name)

We are going to use a pre-trained word vectors. Link these word vector files into your directory (don't copy because these are pretty big)

`/project/smtstud/ss18/data/vec/Wemb.en.filtered.lowered`
`/project/smtstud/ss18/data/vec/GoogleNews-vectors-negative300.bin`
You can link using this command (the final dot means "the current directory")
`ln -s /project/smtstud/ss18/data/vec/Wemb.en.filtered.lowered .`
`ln -s /project/smtstud/ss18/data/vec/GoogleNews-vectors-negative300.bin .`

Later you are going to calculate the cosine similarity of words represented in vector spaces. Copy the script into your directory.

```
/project/smtstud/ss18/commands/vector.py
/project/smtstud/ss18/commands/vector_big.py
```

A. Word Embeddings

1. In the last page, you can find visualization from word embeddings obtained from English and French machine translation data. It is also available at http://i13pc106.ira.uka.de/~echo/research.pdf. Words are filtered under the topic "Research".

   - Find at least three groups where you can see the similarity in meanings and mark on the visualization. You can also check a English-French dictionary: http://enfr.dict.cc/.

2. We are going to examine pre-trained 200-dimensional vectors for 10K English vocabulary. Examine the format of the file.

   `head -n 2 Wemb.en.filtered.lowered | tail -n 1`
   It consists of a word and its vector representation in a 200-dimensional space.

   Word vectors represent semantic and syntactic relationship between words. For example, *lady* should be closely related with *woman*. Check their vector values.

   - Check vector values of the word *lady* and *woman*.

word	v_0	v_1	v_2	v_3	...	v_{199}
lady	0.014	0.047	-0.005	...	0.019	
woman	0.077	0.041	-0.018	...	-0.027	

   We can check the vector values of the two words.

```
lady: 0.01361018 0.04732877 -0.00540839 -0.09349286 ..... -0.05747397
0.01850754
woman: 0.07737311 0.04137088 -0.01775537 -0.13916844 ..... -0.02483073
-0.02679169
```

   As you can see, it is hard for us to understand the vector representation as it is. Therefore, we are going to calculate their distance using a script.

3. We can check the similarity of the words by loading the vectors and calculating the distance between them.

   `python vector.py`

- What is the most similar word to the word *lady*? What is their similarity?
- Find the most similar word for following words and fill the table.

Input word	Most similar word	Similarity
lady	woman	0.63
book	magazine	0.60
mother	mum	0.59
school	college	0.60
great	good	0.49
tree	forest	0.56

- Find the similarity score of following words and fill the table. When is the similarity score high?

word A	word B	similarity
father	dad	0.54
human	animal	0.26
apple	big	0.06
soccer	football	0.63

- Find the word which fits in the semantic relationship.

man : husband = woman :	wife
grass : green = sky :	blue
tree : forest = water :	ocean

4. We can repeat the above experiment using Google's 3 billion vectors.

```
python vector_big.py
```

- What is the most similar word to the word *lady*? What is their similarity?
- Find the most similar word for following words and fill the table.

Input word	Most similar word	Similarity
lady	woman	0.63
book	tome	0.75
mother	daughter	0.87
school	elementary	0.78
great	terrific	0.80
tree	trees	0.83

- Find the similarity score of following words and fill the table. When is the similarity score high?

word A	word B	similarity
father	dad	0.80
human	animal	0.5
apple	big	0.07
soccer	football	0.73

man : husband = woman :	mother
grass : green = sky :	Flares_lit
tree : forest = water :	groundwater
king : queen = man :	woman
paris : france = rome :	italy

- Find the word which fits in the semantic relationship.

Do you notice any difference between two vector spaces ?

B. Neural Language Model

1. Take a look into the log file of training a neural language model.
   `/project/smtstud/ss18/data/rnnlm/Train.log`

   - How is the perplexity score for development data?

   As the training goes on the score sinks down.

2. There are four RNN language models trained. You can find them in
   `/project/smtstud/ss18/models/rnnlms/`

   All models are trained with top 5,000 words.

   (a) Forward LM, two layers
   (b) Backward LM, two layers
   (c) Forward LM, two layers but half of the size
   (d) Forward LM, one layer

   We are going to apply each LM on the test data.

   `/project/smtstud/ss18/data/rnnlm/test.de`

   - Which sentences in the test data should have lower perplexity? Why?

   The sentence with correct verb conjugation should be low (melde ... ab, schlage ... auf, Freund ... ist).

3. Copy the following script into your directory and run it in your directory.

   `/project/smtstud/ss18/bin/rnnlm/Test.back_forward.sh`

   The script calculates perplexity for each sentence using the backward and the forward LM.

   - How is the perplexity for each sentence using each model?
   - Note: we showed the score of each sentence. Perplexity is the negative score (it cannot be negative).
   - Note: lower perplexity means the model is more certain about the sentence.

	Forward	Backward
ich melde mich für die Konferenz an .	30.55	31.03
ich melde mich für die Konferenz auf .	31.72	31.30
ich schlage mit der rechten Hand auf .	43.46	42.85
ich schlage mit der rechten Hand vor .	43.81	44.21
mein Freund , den ich seit vielen Jahren kenne , ist nach Stuttgart gezogen .	78.05	69.99
mein Freund , den ich seit vielen Jahren kenne , sind nach Stuttgart gezogen .	81.13	73.42
die Verkäuferin ist nett .	20.29	20.92
die Marklerin ist nett .	20.29	20.92

   Sentences with correct prepositions have better scores. The word *Verkäuferin* and *Marklerin* are unknown to the language model, therefore the sentences have the same scores.

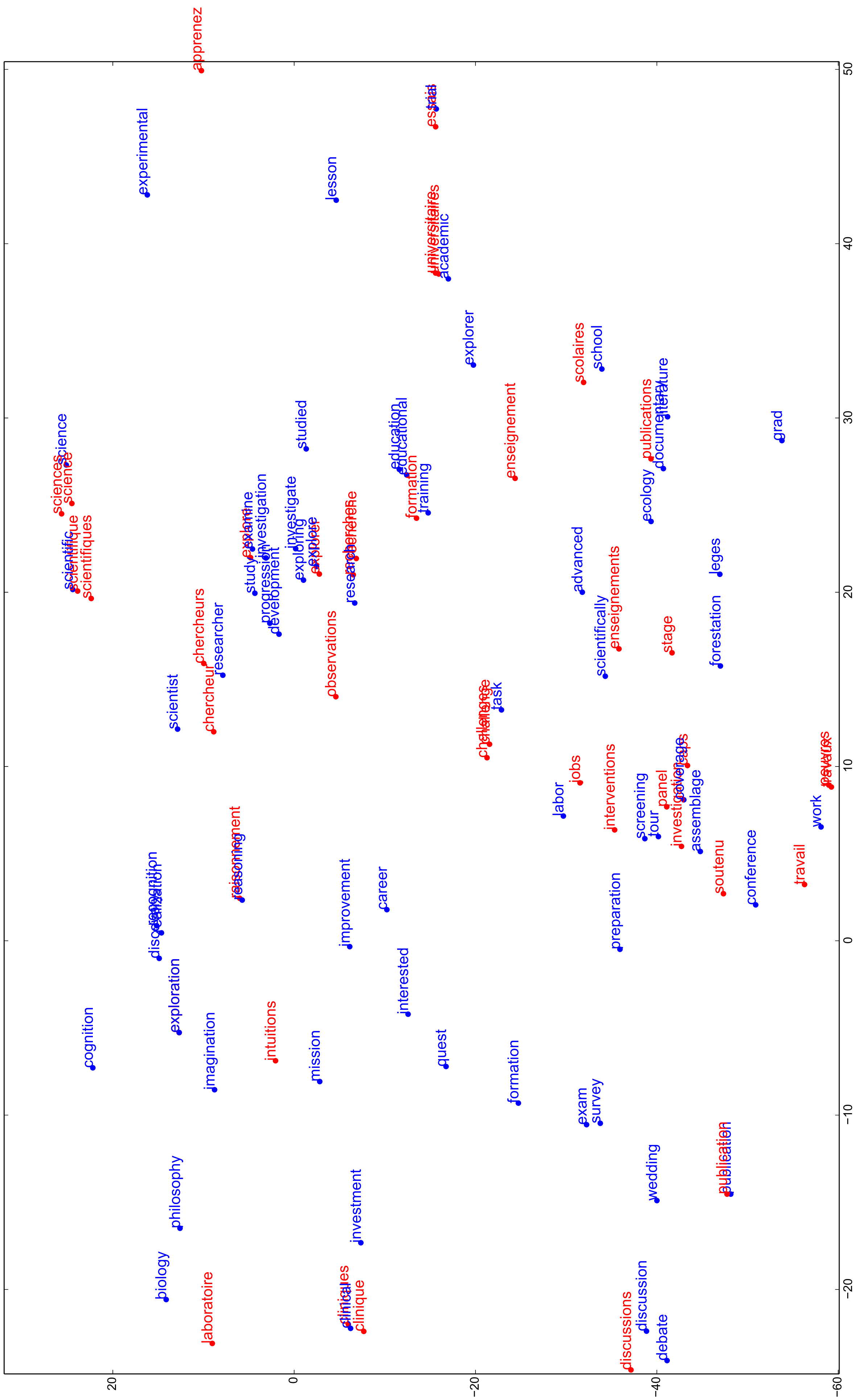
4. For the same test data, try an LM that has half-sized dimensions and another LM that has only one layer. Copy the following script into your directory and run it in your directory.

   `/project/smtstud/ss18/bin/rnnlm/Test.halfdim_onelayer.sh`

   - How is the perplexity for each sentence using each model?

5. Feel free to try your own examples by inputting your own `testdata` in the bash file.

	HalfDim	OneLayer
ich melde mich für die Konferenz an .	28.95	28.64
ich melde mich für die Konferenz auf .	31.04	30.33
ich schlage mit der rechten Hand auf .	40.07	39.35
ich schlage mit der rechten Hand vor .	41.28	40.32
mein Freund , den ich seit vielen Jahren kenne , ist nach Stuttgart gezogen .	77.27	75.88
mein Freund , den ich seit vielen Jahren kenne , sind nach Stuttgart gezogen .	78.24	77.71
die Verkäuferin ist nett .	18.77	20.29
die Marklerin ist nett .	18.77	20.29



apprenez
experimental
lesson
universitaires
academic
explorer
scolaires
school
enseignement
publications
literature
grad
science
sciences
scientifiques
scientific
studied
education
educational
formation
training
investigate
investigation
exploring
explorer
recherche
study
progression
development
research
chercheurs
researcher
chercheur
scientist
observations
task
challenges
advanced
enseignements
scientifically
stage
leges
forestation
ecology
labor
jobs
interventions
screening
panel
assemblage
conference
work
travail
soutenu
raisonnement
reasoning
improvement
career
preparation
interested
exploration
cognition
imagination
intuitions
mission
quest
formation
exam
survey
wedding
publication
philosophy
investment
biology
laboratoire
clinique
discussions
discussion
debate