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

MT Praktikum - Word Embeddings & NNLM

2. Juli 2018

Um die Aufgaben auszuführen, können Sie Ihre Daten in folgendem Verzeichnis speichern:
`/project/smtstud/ss18/systems/USERNAME/`

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. You are provided with 2 word vector files. The *Wemb.en.filtered.lowered* is a set of pre-trained 200-dimensional vectors for 10K English vocabulary, that we will mostly use for small experiments. The other file, *GoogleNews-vectors-negative300.bin* contains 3 billion words, you can use it with the exercise if you want. Examine the format of the file.

   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*.
   - You can do that using the command *grep lady Wemb.en.filtered.lowered*
   - Write out the first 4 and the last values of the vectors in the table.

word	v_0	v_1	v_2	v_3	...	v_{199}
lady				...		
woman				...		

   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.

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		
book		
mother		
school		
great		
tree		

word A	word B	similarity
father	dad	
human	animal	
apple	big	
soccer	football	

- Find the similarity score of following words and fill the table. When is the similarity score high?
- Find the word which fits in the semantic relationship.

man : husband = woman :
grass : green = sky :
tree : forest = water :

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		
book		
mother		
school		
great		
tree		

- Find the similarity score of following words and fill the table. When is the similarity score high?
- Find the word which fits in the semantic relationship.

Do you notice any difference between two vector spaces ?

word A	word B	similarity
father	dad	
human	animal	
apple	big	
soccer	football	

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

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?

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?

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?

	Forward	Backward
ich melde mich für die Konferenz an .		
ich melde mich für die Konferenz auf .		
ich schlage mit der rechten Hand auf .		
ich schlage mit der rechten Hand vor .		
mein Freund , den ich seit vielen Jahren kenne , ist nach Stuttgart gezogen .		
mein Freund , den ich seit vielen Jahren kenne , sind nach Stuttgart gezogen .		
die Verkäuferin ist nett .		
die Marklerin ist nett .		

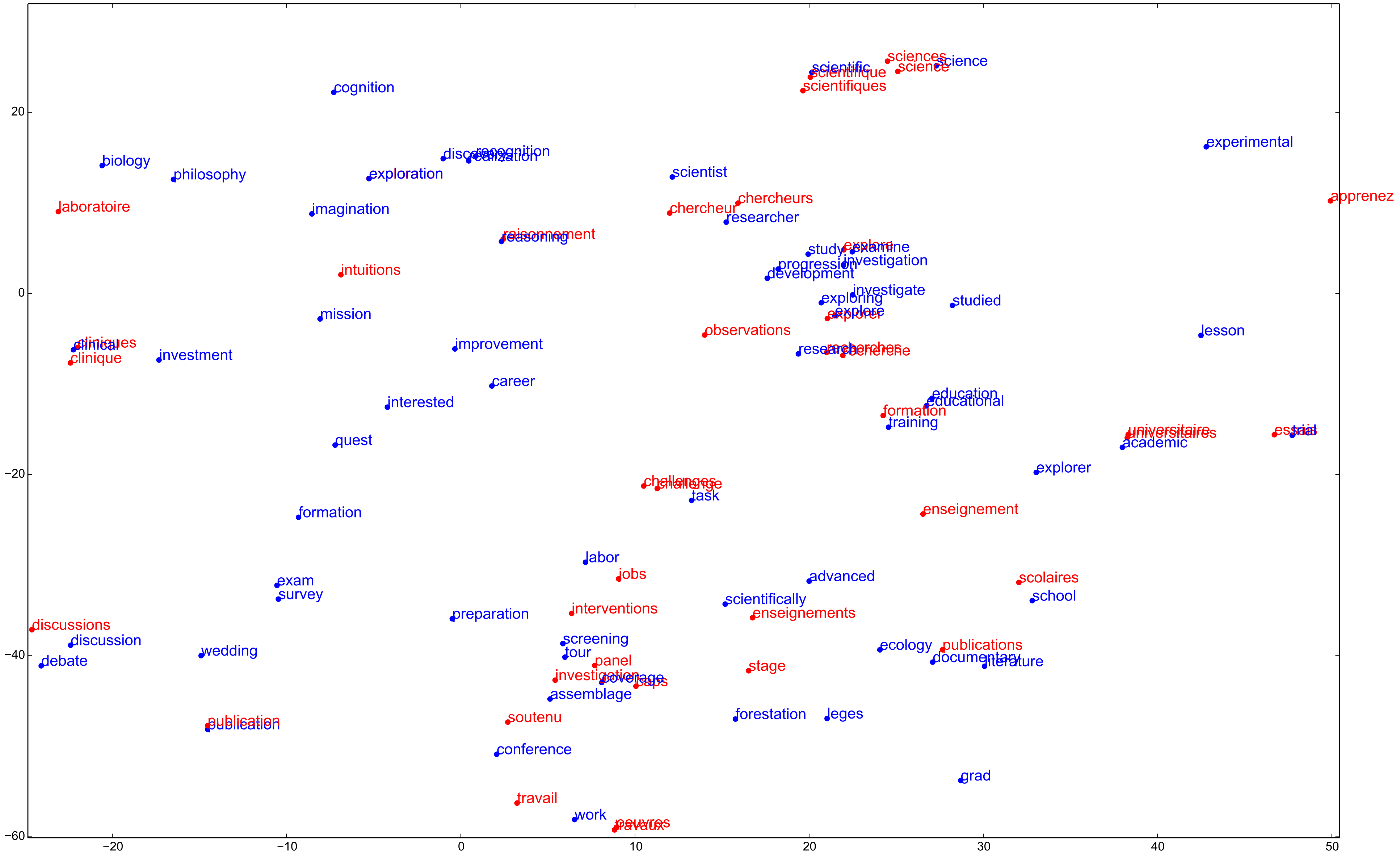
4. For the same test data, try an LM that has half-sized dimensions and another LM that has only one layer. Copy the following script into your directory and run it in your directory.

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

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

	HalfDim	OneLayer
ich melde mich für die Konferenz an .		
ich melde mich für die Konferenz auf .		
ich schlage mit der rechten Hand auf .		
ich schlage mit der rechten Hand vor .		
mein Freund , den ich seit vielen Jahren kenne , ist nach Stuttgart gezogen .		
mein Freund , den ich seit vielen Jahren kenne , sind nach Stuttgart gezogen .		
die Verkäuferin ist nett .		
die Marklerin ist nett .		

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


sciences
science
science
scientific
scientifique
scientifiques
cognition
20
experimental
biology
discovery
recognition
realization
philosophy
exploration
scientist
chercheurs
apprenez
laboratoire
chercheur
imagination
researcher
raisonnement
reasoning
explore
study
examine
progression
investigation
development
intuitions
0
investigate
exploring
studied
mission
explore
explorer
lesson
observations
cliniques
clinical
investment
improvement
research
recherches
recherche
clinique
career
education
educational
interested
formation
training
universitaire
universitaires
essais
trial
academic
quest
explorer
-20
challenges
challenge
task
formation
enseignement
labor
jobs
advanced
scolaires
exam
survey
interventions
scientifically
school
preparation
enseignements
discussions
screening
ecology
publications
discussion
wedding
tour
documentary
debate
-40
panel
stage
literature
investigation
coverage
caps
assemblage
forestation
leges
publication
publication
soutenu
conference
grad
travail
work
oeuvres
travaux
-60
-20
-10
0
10
20
30
40
50