# LIN5580-SEM1-A: Computational Lexical Semantics SEM I

## Year 2016-2017 Final assignment

**How to do the assignment**

The assignment can be done individually or in pairs. The prescribed length is between 12 and 17 pages for an individual assignment; or between 17 and 22 pages for a collaborative effort. The page count does not include references or appendices.

**For a pair assignment, a statement of how the work has been divided among collaborators must be included.** Fifty percent of the assignment, however, can be joint responsibility. The same mark will be assigned to both members of the group.

**Things to look out for**

When writing your assignment, make sure that you: Cite the literature you are using appropriately; Display data and examples using tables and figures, where necessary; Write clearly and concisely.

**Submission**

Upload your answers to the questions in the assignment sheet as a pdf file to the VLE. Upload your code as a python file. More detailed instructions on how to do this will be made available on the VLE page.

**Deadline**

The deadline for submissions is Wednesday 25th of January.

### Intro:

During the last couple of lectures, I discussed distributional semantic models (DSM). Please refer back to the slides of the lectures on DSMs as well as any relevant literature. DSMs are computational models that build semantic representations for words from their contexts in corpus data. In this assignment we will create DSMs using two types of context. I will guide you step by step and will ask you to reply to questions on the way. The questions are numbered. Please give answers to these questions in your report. Your submission on the VLE should include the report as a pdf file and a piece of code in Python.

1) Explain how the Distributional Hypothesis underlies the assumption that distributionally similar words are semantically similar.
2) Because DSMs generate ranked lists of semantically similar words, they can be seen as alternatives to manually built lexical resources such as WordNet. Explain

the main differences between the lexical semantic content of WordNet and the lexical output of DSMs and discuss advantages and drawbacks of manually built lexical resources such as Wordnet and automatically generated lexical resources such as lists of semantically similar words stemming from DSMs.

In this assignment you are going to gather semantically similar words using the following types of data:

1) A monolingual corpus. You can use the corpus files that are provided as part of the DISSECT package (or the Wordspace project), for English. You can also choose to extract data for a language of your choice from a corpus of your choice. I will give you extra points if you do.

You can find the data for DISSECT here:
http://clic.cimec.unitn.it/composes/toolkit/exercises.html

Co-occurrence counts for nouns, verbs are extracted from Wikipedia, BNC and ukWaC corpora (core.sm). The two files core.rows, core.cols contain the lists of words and contexts, respectively

2) Word aligned parallel texts for a language pair of your choice (Do make sure that the language you have chosen under 1) is one of the languages in the pair).

In order to gather semantically similar words, we need A) an *extraction program* (for the extraction from the parallel text and, if you have chosen a different language under 1), for the monolingual text as well)*:* a program that gathers coocurrence counts for the target words and constructs a so-called cooccurence matrix, and B) *a DSM tool,* a program that will compute the similarity between the coocurrence vectors for the head words we are interested in – taking the coocurrence matrix as input.

You will need to write the *extraction program* in Python[1]. You will not need to program the *DSM tool*. For the latter, we will use available software instead.

## Building a coocurrence matrix from a corpus:

There are several ways to build a coocurrence matrix depending on the target words you are selecting (parameter 1) and the type of features, more in particular, the type of context you are selecting (parameter 2). **Please refer back to the slides of the lecture on DSMs for a list of parameters and a description of the options.**

### TARGET WORDS:

3) Do the monolingual texts you are using to extract the cooccurrence counts from provide lemmas? Discuss why using lemmas instead of words can lead to better performances in DSMs.

1 You can use the code you wrote during the practicals for the extraction from the aligned parallel text.

4) Extra points: The parallel text does not provide lemma information, nor PoS tags. Run a PoS tagger on (both sides of) the parallel text and use the information it gives you for the DSM.

The data provided with DISSECT already selected target words for you. For the parallel text, and in case you selected your own monolingual text, we take the 1550 most frequent content words (lemmas) in the corpus as target words. In order to determine the 1550 most frequent content words in either corpus, you will need to count all content words. (If you do not have access to PoS information, use a list of stopwords that you filter out.) You then need to rank these words according to their frequency and take the top 1550.

5) Explain why PoS information can be useful for distinguishing between ambiguous terms. Give some examples from the data you are using.

## FEATURES:

6) Please discuss in your report, the different types of features, more in particular, the types of context (parameter 2) that are employed for DSMs in general.

For the aligned parallel text, we are going to select the words that are aligned to the target word as features. Please select only the 10000 most frequent content words as features.

If you opted to include your own monolingual corpus, we are going to select a small window of the content word following and the content word preceding the target word, as features. Please select only the 10000 most frequent content words as features.

7) Discuss what influence the size of the window has on the nature of the semantic relations we find between target word and semantically similar words proposed by the system. What consequences do you think a larger window size would have on the semantically similar words proposed by the system?
8) Discuss the differences in output of the DSM (in terms of the similar words it will produce) you expect between the two types of input we are using (the monolingual corpus and the aligned parallel files). Which type of data do you think will generate the highest percentage of synonyms amongst the nearest neighbours?
9) Write a python program that extracts a sparse coocurrence matrix from the parallel aligned files (and the monolingual corpus, if you are using a different corpus than the one provided by DISSECT). Include the extraction program you wrote in the submission, and paste 20 lines of the sparse coocurrence counts you extracted in the report.

For **Wordspace**, the format should be a file with three columns, separated with tabs. If you downloaded the most recent version of Wordspace from R-Forge, you will find an example file here: <path_to_wordpace_library>/wordspace/extdata/term_context_triplets.gz

The first column contains the lemma/word the second the feature and the last the coocurrence count.

Dog <TAB> eat <TAB> 4 (for eat following dog) eat <TAB> dog <TAB> 2 (for dog following eat) Reading in the data for Wordspace works as follows:

MyTripletDSM <-
read.dsm.triplet("<path_to_wordpace_library>wordspace/extdata/term_context_triplets.gz", freq=TRUE)

For **DISSECT**, you can find information on input formats here:
http://clic.cimec.unitn.it/composes/toolkit/matrix_file.html#cooccurrence-matrix-file

## Feature weighting and similarity function:

We discussed two additional parameters of DSMs in class: the feature weighting scheme and the similarity function.

10) Explain what function the feature weighting scheme has. What would be the drawback of doing without a weighting scheme and using raw frequencies instead?
11) List the weighting schemes the tool you are using includes and what similarity functions. Make your choice for the weighting scheme and similarity function you will use and motivate your choice in the report.

One of the weighting schemes **DISSECT** includes is Positive Pointwise Mutual Information (similar to what we discussed in class: http://clic.cimec.unitn.it/composes/toolkit/creating.html ).

For **Wordspace**, refer to http://wordspace.r-forge.r-project.org/pdf/coling2014_wordspace_final.pdf to find out about the weighing schemes( association measures) it offers.

One of the similarity functions available in **DISSECT** is the Cosine similarity (http://clic.cimec.unitn.it/composes/toolkit/querying.html). We discussed it in class.

Check the similarity measures **Wordspace** here:
http://rpackages.ianhowson.com/rforge/wordspace/man/dist_matrix.html

## Analysing the output:

Now we are ready to generate some output. You should be able to get the distributional similar words of any of the 1550 target words you selected.

For **DISSECT**, see 'Finding neighbours' in http://clic.cimec.unitn.it/composes/toolkit/querying.html .

For **Wordspace** see example code on: http://wordspace.r-forge.r-project.org/pdf/coling2014_wordspace_final.pdf and http://rpackages.ianhowson.com/rforge/wordspace/man/nearest_neighbours.html

12) For both types of corpora (multilingual and monolingual), retrieve the top-5 semantically similar words for five high-frequency words and five low-frequency words from the 1550 words. Paste the output in the report. Is there a difference in the quality of the output for the high-frequency versus the low-frequency words? Is there a difference for the two types of corpora? Can you explain what you see?
13) Explain the notion of contextual variability. Take a polysemous word of your choice that is among the 1550 head words you selected. Generate the top-10 semantically similar words for this target word using the monolingual DSM you created. Paste the output in the report. Do you see some consequences of contextual variability in the output for this word? Now take a pair of homonyms and do the same. What are your observations?

## Compositionality:

We discussed how DSMs can cater for compositionality in week 9.

14) Explain the principle of compositionality and discuss a couple of different ways in which DSMs have been adapted to cater for compositionality.
15) Now, let us experiment with compositionality in the monolingual DSM you have just built. Follow the instructions on http://clic.cimec.unitn.it/composes/toolkit/composing.html

    and select a model of your choice (motivate your choice). Show some example output of the composed DSM. Analyse the results.

16) Is it possible to create a compositional model for the multilingual DSM with the models you have chosen? If so, show some example output here as well. Analyse the results.

## Relating things back to theory:

We discussed several theories in class that try to explain the lexical semantics of words. We are now trying to couple our knowledge from one of these theories to the things we observe in the DSM.

17) Discuss the basic tenet of the Prototype Theory in lexical semantics and discuss its advantages when compared to the classical theory in terms of its explanatory power.
18) Does the DSM cater for typicality effects? (Hint: You can check the distributional similarity of types of fruit and see if the prototypical examples are more semantically similar to each other than less prototypical examples. You can also check whether prototypical types of fruit have more statistically prominent properties in the coocurrence matrix than less prototypical types of fruit.)