NTMI - Project Exercises - Part A

Alex Khawalid (10634207)
Wessel Klijnsma (10172432)
Winand Renkema (10643478)

February 4, 2016

Step 1: Extracting n-gram statistics

1 Introduction

This is a report about the first assignment of the Natural Language Models and Interfaces. Its goal is to build a program that is able to create tables of n-gram frequencies based on a corpus. N-grams are word sequences of length n that form the basis of many probabilistic language models such as Markov Models

2 Method

The program is written in the Python programming language. For this assignment the novel Emma by Jane Austen will be used as a corpus. However other corpora can be used as input of the program. The approach used to create the program is to loop through each word of every sentence in the corpus. For each word also the next n-1 words are taking into account. If the sequence occurs in the table its frequency is incremented by 1. If not, a new table entry is created. The frequency table is represented by a hash-table datastructure, where the n-gram is the key and the frequency its value. In order to get the m most frequent n-grams, the hash-table is sorted on the frequencies and the first m elements of the array are the m most frequent n-grams. For this assignment the 10 most frequent n-grams are determined for n=1, n=2, and n=3. The results can be found in the results section.

3 Results

Frequencies for $n = 1$ and $m = 10$:

rank	*unigram*	*frequency*
1	the	20829
2	to	20042
3	and	18331
4	of	17949
5	a	11135
6	her	11007
7	I	10381
8	was	9409
9	in	9182
10	it	7573

The sum for $n = 1$ and $m = 10$ is 617091.

Frequencies for $n = 2$ and $m = 10$:

rank	*bigram*	*frequency*
1	of the	2507
2	to be	2233
3	in the	1917
4	I am	1366
5	of her	1264
6	to the	1142
7	it was	1010
8	had been	995
9	she had	978
10	to her	964

The sum for $n = 1$ and $m = 10$ is 617090.

Frequencies for $n = 3$ and $m = 10$:

rank	*trigram*	*frequency*
1	I do not	378
2	I am sure	366
3	in the world	214
4	she could not	202
5	would have been	189
6	I dare say	174
7	a great deal	173
8	as soon as	173
9	it would be	171
10	could not be	155

The sum for $n = 3$ and $m = 10$ is 617089.

The program can be run using the following commands:

```
./a1-step1 -corpus [path to corpus] -n [length of n-gram] -m [limit of the rank of the most
frequent n-grams]
```

4 Discussion

Looking at the results, one can see that the n-gram consist mostly of small, simple words such as determiners, pronouns and verbs like: be, could, would and have. Also signs of the effect of Zipf's law are present, 'the' (rank 1 for 1-grams) occurs about 2.75 times more often than 'it' (rank 10 for 1-grams).