ASSIGNMENT 2

Subject : Spelling Correction
Handed Out : 14.03.2018
Due date : 03.04.2018

Please submit your solution (code and a README file) by 17:00 pm on the due date. Please describe your code in detail in README file.

Introduction

Spelling correction takes an input text and returns a *corrected* form that is crucial in Natural Language Processing (NLP) tasks. For example, ”compter” is transformed into ”computer”. We try to correct spelling errors by using Hidden Markov Models (HMMs) [1] in this assignment.

0.1 Hidden Markov Models (HMMs)

You will use HMMs for spell correction in this assignment. In an HMM, there are hidden states and observed states. In this experiment, your hidden states will be the misspelled words and your observed states will be correct forms .

0.1.1 Build hidden Markov Model

- The initial probabilities $p(w_i)$: the probability that a sentence begins with a correct word w_i
- The transition probabilities $p(w_{i+1}|w_i)$: the probability that word w_{i+1} is seen after the word w_i
- The emission probabilities $p(x|w_i)$: the probability that correct form of misspelled word x is w_i

To calculate emission probabilities, firstly we should know the correct form of the misspelled word. The problem is that the number of possible correct form of the misspelled word is infinite. To define the possible correct forms of the misspelled word, we use the minimum edit distance. If distance $d = 1$, we add it to list of possible correct forms of the misspelled word. Example of misspelled and correct form of a word is given in Table 1.

We calculate the deletion, insertion and substitution dictionaries by using Formula 1.

Table 1: Candidate corrections for the misspelling *acress* and the transformations that would have produced the error.

Error	**Correction**	**Correct Letter**	**Error Letter**	**Type**
acress	actress	t	-	deletion
acress	cress	-	a	insertion
acress	access	c	r	substitution
acress	across	o	e	substitution
acress	acres	-	s	insertion

$$\begin{aligned}
del[a,b] &= count(ab\,typed\,as\,a)\;where\;w\;has\;w[j-1]=a\;w[j]=b,\;x\;has\;x[j-1]=b \qquad (1)\\
&\;where\;w[j]\;is\;the\;letter\;in\;the\;jth\;position\\
ins[a,b] &= count(a\,typed\,as\,ab)\;where\;w\;has\;w[j-1]=a,\;x\;has\;x[j-1]=a\;x[j]=b\\
sub[a,b] &= count(a\,typed\,as\,b)\;where\;w\;has\;w[j]=a\;,\;x\;has\;x[j]=b
\end{aligned}$$

Then, we will calculate $p(x|w)$ bu using Formula 2 based the deletion, insertion and substitution dictionaries.

$$P(x|w_i) = \begin{cases} \frac{del[a,b]}{count[ab]} & , if\ deletion \\ \\ \frac{ins[a,b]}{count[a]} & , if\ insertion \\ \\ \frac{sub[a,b]}{count[a]} & , if\ substitution \end{cases} \tag{2}$$

You will calculate transition probabilities (bigram language model) from the corrected dataset.

0.1.2 Viterbi

Your Viterbi algorithm will find the best possible correct form of the misspelled word by looking all the possible sentences. Your Viterbi algorithm will consist of two steps:

1. Youl will compute the probability of the most likely word sequence.
2. You will trace the back pointers to find the most likely sentence from the end to the beginning.

An example Viterbi trellis is given in Figure 2.

0.1.3 Evaluation

Your program will compute the accuracy of the implemented spelling correction model as the ratio of the correctly found forms of the misspelled words to the total number of misspelled

Figure 1: Viterbi

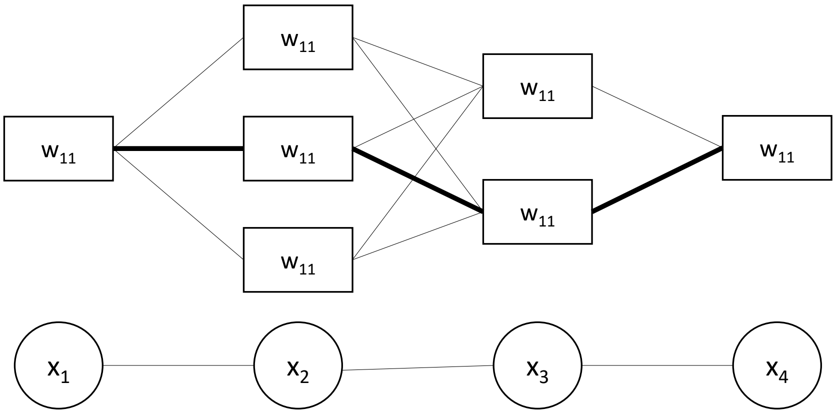
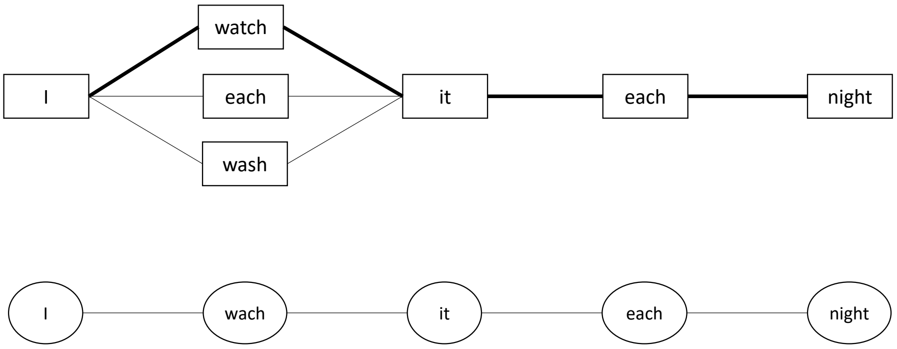


Figure 2: Example of Viterbi



words as given in Formula 3.

$$A(W) = \frac{\#\, of correct\_found\_words}{\#\, of total\_misspelled\_words} \tag{3}$$

Dataset The dataset contains sentences in it. Some words are miswritten but their correct forms are provided.

```
I have four in my Family Dad Mum and <ERR targ=sister> siter </ERR> .
```

You need to use regex to extract the correct form of the misspelled word and swap the correct form of the misspelled word with the correct form. Therefore, you will have a clean dataset to build a language model. You will use cleaned dataset to compute the transition probabilities.

To download the dataset, please click the link :
Dataset : `https://piazza.com/class_profile/get_resource/jdiv1jqrgrp7kl/jeebzpd1rzj6oa`

Notes

- Do not miss the submission deadline.
- Compile your code on *dev.cs.hacettepe.edu.tr* before submitting your work to make sure it compiles without any problems on our server.
- Save all your work until the assignment is graded.
- The assignment must be original, individual work. Duplicate, very similar assignments or code from Internet are going to be considered as cheating.
- You can ask your questions via Piazza and you are supposed to be aware of everything discussed on Piazza. You cannot share algorithms or source code. All work must be individual! Assignments will be checked for similarity, and there will be serious consequences if plagiarism is detected.
- You need to implement either in **Java** (Java 1.8) or **Python** (Python 3). Please submit your source codes and README file in the following submission format.
- You will be graded not only for the output, but also readibility, comment lines and README.md.
- I will run your programs from the command line as following. Any other command line format will not be accepted!
  Python

  ```
  python3 assignment2.py dataset.txt out.txt
  ```

  Java

  ```
  Java Main dataset.txt out.txt
  ```

```
→ <studend id>
    → code.zip
    → README.md
```

References

[1] Daniel Hladek, Jan Stas, and Jozef Juhar. Unsupervised spelling correction for slovak. *Advances in Electrical and Electronic Engineering*, 11(5):392, 2013.