Assignment 2: Neural Language Models

Submission: November 9th 2017 (before the end of the day)

Advanced Natural Language Processing
COSC 7336 - Fall 2017

1. **Character based-language model practical:**
   Complete the Jupyter Notebook from this link.

2. **Text Summarization:**
   There is a lot of practical value in the ability of automatically summarizing text. For the second part of this practical, you will need to design and train a system to perform text summarization.

   - **Method**
     You must design and implement a recurrent-neural-network-based language model able to generate the summary from the input text.
   - **Data sets**
     For this assignment, the test datasets are transcripts from TED talks and news articles from the Gigaword English corpus. For the TED talks the goal is to generate a summary from the transcript. For the news articles, the goal is that your summarizer model will generate the title of the article. In Table 1 we show examples of the input and desired output for each data set.

     We do not expect that you will design two different models. Rather, we want you to focus on the news articles, as we provide much more data for training and parameter tunning and then observe how that model performs on the TED talks.
   - **Prediction output**
     In order to run the evaluation scripts, your program output has to conform to this format:

     ```
     filename1.txt, summary for file 1
     filename2.txt, summary for file 2
     filename3.txt, summary for file 3
     ```
   - **Evaluation of Summarization Systems**
     Summarization systems are usually evaluated in terms of how well the automatic summary correlates with the reference summary. Here we will use BLUE to score your system submission. BLUE is a measure used commonly to evaluate machine translation systems, but it will be a good alternative in this task. BLUE measures correlation between the reference summary and the generated summary in a precision oriented manner. The formula for the BLUE metric is as follows:

     $$BLEU = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{1}$$

     where BP is a penalty for small translations, and p_n , n-gram precision, is computed as shown in Equation 2 below.

     $$p_n = \frac{\sum_{C\in\{Candidates\}} \sum_{n-gram\in\{C\}} Count_{clip}(n-gram)}{\sum_{C\in\{Candidates\}} \sum_{n-gram\in\{C\}} Count(n-gram)} \tag{2}$$

Corpus	**Input Text**	**Summary**
Gigaword	Ground employees of Philippine Airlines (PAL) Friday defied a government order to end a three-day strike that has stranded thousands of passengers. The strike was mounted Wednesday over wage increases and job security.	Philippine Airlines workers refuse to end strike
TED Talks	Here are two reasons companies fail: they only do more of the same, or they only do what's new. To me the real, real solution to quality growth is figuring out the balance between two activities: exploration and exploitation. Both are necessary, but it can be too much of a good thing. Consider Facit. I'm actually old enough to remember them. Facit was a fantastic company. They were born deep in the Swedish forest, and they made the best mechanical calculators in the world. Everybody used them. And what did Facit do when the electronic calculator came along? They continued doing exactly the same. In six months, they went from maximum revenue ... and they were gone. Gone. To me, the irony about the Facit story is hearing about the Facit engineers, who had bought cheap, small electronic calculators in Japan that they used to double-check their calculators. (Laughter) Facit did too much exploitation....	Is it possible to run a company and reinvent it at the same time? For business strategist Knut Haanaes, the ability to innovate after becoming successful is the mark of a great organization. He shares insights on how to strike a balance between perfecting what we already know and exploring totally new ideas – and lays out how to avoid two major strategy traps.

Table 1: Example of input and desired output for the Gigaword and TED talk corpora. To save space, we only show a fragment of the transcript of the TED talk.

Where $Count_{clip}(n-gram)$ is the maximum number of overlapping n-grams in the reference summary and the candidate summary, and $Count(n-gram)$ is the number of n-grams in the candidate summary.

- **Download Data**
  The TED talk data for this assignment is in this link TED Talks. Note that we have removed the summaries from the test set (instances 1669–2025).

  The Gigaword data is in this link: Gigaword. You have three disks in there. The test data is labeled as test data and only has the content of the newswire, not the headlines. The other two disks are provided for your training of models. You are free to use both of them or only one or part of one. Given the size of the data, you may run into space issues. Please let us know (send us an email) if you need a virtual machine in Azure to complete this task.

- **CodaLab Submissions:** We have set up this assignment as a shared task in CodaLab. Instructions for this are in the handout for assignment 1. Access to the shared task is through this link (or search for text summarization in Coda Lab). Please make sure you're regularly checking this link, as we may need to change it if we find that we have to make changes to the shared task set up.

Note that we have set up two phases, one per test data set. For successfully completing your system submission you need to submit a text file (submission.txt) with your system's predictions and your source code in a single zip file for each phase by the submission deadline. The submission.txt file should contain one instance prediction per line and it should begin with the file name, as discussed above. It's important that the zip file only contains the two files mentioned above, not a zip containing a folder with the contents in it; this will cause the evaluation script to fail.

NOTE: CodaLab is an open source framework for running competitions. Your system submissions will be ranked according to accuracy of the system but the ranking will be public and thus it's super important that the username you choose for the submission is not disclosing your identity. In order to identify which student gets credit for which system submission, please note in your report, and in your source code, the name you used to identify your submission in CodaLAb (user name).

Deliverables

You will need to turn in the following:

1. The Jupyter Notebook with the appropriate exercises completed via the file request in Dropbox.
2. System predictions in CodaLab
3. System submission in CodaLab
4. Technical Report describing the system and relevant experiments and analysis.
5. Zip file with source documents and report. Create a zip file with all the above items and submit using Dropbox file request. Name your zip file `cosc7336-assign2-student-name`.zip.

Resources

Some useful links:

- Text summarization with RNNs