The Study Of The Effect Of Training Set On Statistical Language Modeling

Xipeng Shen, Bo Xu
National Laboratory of Pattern Recognition, Institute of Automation
Chinese Academy of Sciences, Beijing, P.R. China
{xpshen, xubo}@nlpr.ia.ac.cn

Abstract
In this work, we make a study on the effect of training set on statistical language modeling (SLM). A corpus selection system based on perplexity is presented. It is tested in two experiments: one is to select optimal training corpus for generating a domain-specific SLM; the other one is for generating an optimal SLM for a LVCSR system. The results show that the training corpus is important for the capability of SLM and our corpus selection system is powerful for optimal corpus selection. With the help of this system, we generated a SLM for a LVCSR system, which contributed 14.5%--17.7% relative character error reduction.

1. Introduction
Statistical language modeling (SLM) is crucial for a large variety of language technology applications, such as speech recognition, information retrieval, handwriting recognition, spoken language understanding and so on [1]. As this kind of language model is developed from text corpus, the training set plays an important role in the capability of the SLM. Some works on the selection of training set have been done recently, such as [2], [3], [4], [5]. Particularly, in [2], the author suggested to use a log-likelihood based criterion to select articles from a training corpus that are suitable to reduce perplexity on a specific task defined by a small target corpus. However, those previous works didn’t present the direct effect on the recognition rate (RR). In [2], the author adopted one method to remove articles from the background corpus. While in our work, the opposite method is adopted—to add more and more articles from the background corpus. The other feature in our work is that the unit of corpus to be added into the training set is more refined. We calculated each paragraph’s perplexity to judge whether to add this paragraph. We studied the effect of training set on SLM from two chief aspects—one is for a specific task and the other one is for a LVCSR system.

In the following sections, first we outline the frame used to select the optimal training corpus in Section 2. Then, in Section 3, we present our experimental results, which include two chief parts. One is to select optimal corpus for a specific task defined by 10,000 sentences. In this experiment, we took use of a corpus with 30M bytes as the seed set and the 10,000 sentences as test set. Through the experiment, we investigated the effect of our selection method on specific tasks. In the other experiment, we applied this method to a LVCSR system. Perplexity and RR are presented as the experimental results. In section 4, discussion and several conclusions on the experimental results is proposed.

2. Frame for training set selection
To select the optimal corpus, we built the following frame based on criterion of perplexity. Our chief idea is as following: first, prepare a seed set and a test set according to the task; secondly, calculate the perplexity of each paragraph of background corpus based on the SLM trained by the seed set; thirdly, add those paragraphs with smallest perplexity into the seed set and train a new SLM; then, go to the second step to iterate those steps until meeting the end condition. In our actual work, three preparations were made before the chief process. First, we preprocessed the raw corpus in order to clean the OOV (Out Of Vocabulary) characters and regularize the raw corpus. This is to prepare for building index files and other works. Then, we segmented all of the words and represented those words with Word No. Thirdly, we built the index files to record the position of start and end of each paragraph. In the factual work, we combined some paragraphs that are composed of too few words.

The following is the main steps in the chief selection process, and figure 1 shows the frame.

1. A seed set and a test set are selected out manually according to the task.
2. Iterative steps to select the optimal training paragraphs:
   - Generate the mid-language model (MLM) based on the training set selected out i.e. the seed corpus;
   - Calculate the perplexity of test set and judge if the end condition is satisfied;
   - If the end condition is satisfied, end the selection process. Otherwise, calculate the perplexity of each remaining paragraph of the background corpus using the MLM.
   - Select the optimal paragraphs whose perplexities are in the least m ones and add them into the seed set.
   - Return to the beginning of the iterative steps.
In our work, the selection criterion to select paragraphs is the bigram perplexity. To calculate the perplexity of each paragraph of the background corpus in the iterative process, the index of the paragraphs of the overall corpus is built before the selection process. Those too short paragraphs are combined into their neighbor paragraphs. The selection of the original seed training corpus and the test set mainly accords to the task. In the experiment for specific task, we selected a corpus of 30M bytes as seed set and 10,000 sentences in a specific domain as the test set. While, for the task of generating a LM for our LVCSR system, we select 15M seed data and 15M test data both including texts in 13 domains. The experimental results are showed in the following sections. We take the perplexity as the criterion for selecting the optimal corpus in the iterative steps. The perplexity of the test set got in every circle of the iterative process is recorded in a history file. The time to end the process is when the perplexity of the test set is \( k \) more than the least perplexity having been obtained or there are no remaining paragraphs in the background corpus. The reason to adopt such end condition is that if the background corpus is not very suitable to the task, the perplexity of the test set will reach its lowest point before all the paragraphs in the background corpus are selected out. While, if the background corpus is small and very suitable to the task, then the lowest point may be reached when all paragraphs in the background corpus are selected out. Our experiments include the following two experiments. One is to generate an optimal LM in a specific domain, and the other one is to generate an optimal LM for a LVCSR system. The experiments and results are showed in section 3.

3. Experiments

3.1 Overall corpus

Overall, there are about 10G text corpus--MarData in our lab. This corpus includes files on various domains, such as news, military, science, religion, sports, literature and so on. Additionally, we downloaded a text corpus of about 30M bytes from the Internet and classified them into 13 domains. This corpus is named as Data30. It seems that these 13 domains can cover a large part of domains in the world. We took use of corpus MarData as the background corpus for our experiments.

3.2 Experiment 1—specific task experiment

3.2.1 Problem

In this experiment, we extracted 10,000 sentences from the corpus Data30 and utilized them as our test set. The object of this experiment is to generate an SLM, which is optimal to the domain composed of sentences like the 10,000 sentences. And also, fairly good results should be obtained to other sentences that are irrelative with the 10,000 sentences. Through this experiment, the effectiveness of the corpus selection process is proved primarily.

3.2.2 Corpus in the experiment

In this experiment, we took use of corpus MarData as the background corpus. We took use of the remaining corpus of Data30 after extracting out the 10,000 sentences. Thus, this seed set is correlative to the test set. We recorded two test speech corpora: Corpus99
and Corpus98. These two corpora both include 10 males' speech and each speaker spoke 60 sentences. The text content of Corpus99 is 600 sentences included in the 10,000 sentences. Text content of Corpus 98 also includes 600 sentences, which are selected randomly and have no relations with the 10,000 sentences and corpus Data30. Our expected result is that the RR of the Corpus99 is very high, and meanwhile, the RR of the Corpus98 is fairly high.

3.2.3 Experiment results

Through the corpus selection system, we obtained an optimal corpus—CorpusO1, which is composed of files of about 368M bytes. We added the 10,000 sentences into CorpusO1 to get another corpus—CorpusO2. And, we also selected 3 corpora (CorpusR1—CorpusR3) from corpus MarData randomly, each of which is of 368M bytes. By CorpusR1—CorpusR3, three SLMs were generated (SLM1—SLM3). SLM4 is a SLM trained using CorpusO1, and SLM5 was trained using CorpusO2. The recognition correct rates for Corpus98 and Corpus99 are showed in table 1. The CER (Character Error Reduction) and relative CER of SLM4 and SLM5 compared to SLM1—SLM3 are showed in table 2.

3.3 Experiment II—to generate a SLM for a LVCSR system

3.3.1 Problem

In this experiment, our objective is to generate an optimal SLM for our LVCSR system. Since Experiment I showed the effectiveness of the corpus selection system in specific task, we implemented this experiment to test whether this system is helpful to LVCSR system.

3.3.2 Corpus in the Experiment

The sentences input into a LVCSR system may be sentences from any domain. Selection of the seed corpus plays an important role in the corpus selection process for a LVCSR system. In our work, we split corpus Data30 into 2 parts: each with 15M bytes and each covers all of the 13 domains. One part is used as seed set and the other one is the test set. Since these two set are from one big corpus, they are similar and relative to each other. The background corpus is still corpus MarData. When we obtained the final optimal corpus and generated the final optimal SLM, we took corpus98 and corpus99 as the test speech corpus to test the RR of the new SLM.

3.3.3 Experiment results

To get the optimal LM, first, we manually select a subset—subMarData with about 5G bytes from corpus MarData as our new background corpus. To accelerate the process of optimal corpus selection, we split corpus subMarData into five subsets, each of which includes about 1G bytes. So, we can start 5 processes to select the optimal corpus from corpus subMarData simultaneously. To test the effective of the selection process for LVCSR system, we recorded the character perplexity (CP) of the test set based on the mid-LMs during one of the selection processes, which are showed in figure 2 and figure 3. There are two curves in both figures. As noted in the figures, the lower curve denotes the CP based on the mid-LMs generated during the selection process. While, the other curve denotes the CP of the test set based on the SLMs that are trained on corpora selected randomly from corpus subMarData.

In figure 2, the mid-LMs were not pruned at all. Each

**Table 1.** Task-specific experiment results

<table>
<thead>
<tr>
<th>SLM</th>
<th>Results for Corpus98</th>
<th>Results for Corpus99</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLM1</td>
<td>73.80</td>
<td>63.08</td>
</tr>
<tr>
<td>SLM2</td>
<td>73.17</td>
<td>62.67</td>
</tr>
<tr>
<td>SLM3</td>
<td>75.07</td>
<td>65.38</td>
</tr>
<tr>
<td>SLM4</td>
<td>75.62</td>
<td>68.99</td>
</tr>
<tr>
<td>SLM5</td>
<td>75.72</td>
<td>71.58</td>
</tr>
</tbody>
</table>

**Table 2.** Comparison of recognition results

<table>
<thead>
<tr>
<th>LM</th>
<th>CER(%)</th>
<th>Relative CER(%)</th>
<th>CER(%)</th>
<th>Relative CER(%)</th>
<th>CER(%)</th>
<th>Relative CER(%)</th>
<th>CER(%)</th>
<th>Relative CER(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>1.82</td>
<td>6.9</td>
<td>1.92</td>
<td>7.3</td>
<td>5.91</td>
<td>16.0</td>
<td>8.5</td>
<td>23.0</td>
</tr>
<tr>
<td></td>
<td>2.45</td>
<td>9.1</td>
<td>2.55</td>
<td>9.5</td>
<td>6.32</td>
<td>16.9</td>
<td>8.91</td>
<td>23.9</td>
</tr>
<tr>
<td></td>
<td>0.55</td>
<td>2.2</td>
<td>0.65</td>
<td>2.6</td>
<td>3.61</td>
<td>10.4</td>
<td>6.2</td>
<td>17.9</td>
</tr>
<tr>
<td></td>
<td>1.61</td>
<td>6.1</td>
<td>1.70</td>
<td>6.5</td>
<td>5.28</td>
<td>14.4</td>
<td>7.87</td>
<td>21.6</td>
</tr>
</tbody>
</table>
time, we added only 30,000 paragraphs to the training corpus, which are about 4.6M in size. It is apparent that the CP of the former curve declines faster than that of the latter one. In figure 3, the mid_LMs were pruned to reduce the size of the mid_LMs. In this experiment, we added small size of corpus into the training corpus in the first 7 iterative processes. It is in order to make the mid_LM grow robust enough to avoid serious excursion in the following process. Then, corpora of large size, about 90M each time, were added to the training corpus. From the comparison, we can see that the corpus selection system has significant effect on the capability of the SLM.

Finally, we generated an optimal SLM—LM1 from all the selected corpora, which include five subsets selected respectively from five processes. Its size is about 1G bytes. Meanwhile, we generated another two SLMs—LM1 and LM2 from two training set selected randomly, which are about the same size as the training set of LM1. Table 3 shows the results of RR for the two test speech corpora—Corpus98 and Corpus99. We can see that LM1 is better than LM2 and LM3. For Corpus98, which is out of the training set, we obtained 14.5%-17.7% CER.

4. Discussion and conclusions

From our experiments, we can see that the training corpus is important for the capability of SLM. More training corpus is not bound to generate a better SLM. Our corpus selection system seems to be powerful to select the optimal corpus. However, there are still some problems needing to be resolved. First, the speed to select corpus needs to be accelerated. Secondly, better results may be obtained if we combine the corpus selection method with domain-relative SLM. Since the corpus selection system is more effective in generating domain-relative SLM, if we generate an optimal SLM for each of many domains using the system and combine these SLMs with different weighs according to the test set in recognition, better result is expected to obtain.

5. References