An Internet Agent for Language Model Construction

Peter Wyard **Tony Rose**[*]

BT Laboratories, Martlesham Heath,
Ipswich IP5 3RE, UK.
peter.wyard@bt-sys.bt.co.uk

Dept. of Computing
Nottingham Trent University,
Nottingham NG1 4BU, UK.

Abstract

A software agent is described which is able to take a seed (reference) corpus specified by the user, search the Internet for documents which are sufficiently similar to the seed corpus (as defined by a set of similarity metrics operating at a number of levels in the text), and augment the seed corpus with these documents. The size of the corpus and, hopefully, the quality of the derived language model, are thus progressively increased. The seed corpus may be quite a small collection of transcripts from the application domain, such as may be collected with minimal effort. Preliminary results are given for the perplexity of language models constructed using this approach. Potentially, our method has applications well beyond speech recognition, in corpus-based language processing in general, and document retrieval.

1 Introduction

One of the main problems in constructing a language model for a speech recogniser application is finding sufficient language data that is characteristic of the intended application domain. Traditional methods, such as "*Wizard of Oz*" data collection or building an incremental series of trial systems are time consuming and expensive.

An alternative approach is to combine a small amount of application domain data with a large amount of more general data to form a language model. This may be done by interpolating two separate models, although the standard way of doing this is not ideal. Rosenfeld (1996) discusses a maximum entropy approach to combining the two sources of data, while Vergyri (1995) discusses an alternative approach based on data bleaching. Rudnicky (1995) also addresses the problem of limited domain data. It often seems that there is no substitute for more data of the right kind.

The Internet provides a vast corpus of language data (although most of it is currently typed text, which may make it less than ideal for spoken language systems). The basic approach of our current work is to start with a small corpus of language data from a particular domain, and to search for "more of the same" on the Internet, using appropriate similarity metrics.

This paper describes a software agent which makes use of a combination of similarity metrics to determine whether to retain a particular candidate document which it finds during its search. Retained documents are used to augment the seed corpus. The size of the corpus and, hopefully, the quality of the derived language model, are thus progressively increased. The seed corpus may be quite a small collection of transcripts from the application domain, such as may be collected with minimal effort.

The similarity metrics operate on the title of the candidate document, if there is a title (metric M1), and at the character, word and phrase levels of its contents (metrics M2 to M4), giving a greater robustness than relying on a single metric. The main novelty of this work is the use of multiple similarity metrics, the way in which they are combined (see Section 4), and their use by an Internet search agent for language model construction. The agent may also be used for document retrieval (Rose & Wyard, 1997).

2 The Similarity Metrics

This section describes the four similarity metrics which the agent may use, and how they are combined in the simplest operation of the agent.

M1 takes the reference corpus and derives a word frequency list, also known as a term frequency list (TFL). Each term frequency (TF) is multiplied by its inverse document frequency (IDF) value to

[*] Now at Canon Research Centre Europe, Guildford, Surrey GU2 5YF, UK. tgr@cre.canon.co.uk

derive a TF.IDF value. This is common practice in the field of information retrieval (Salton & McGill 1983), and it is used to give greater weight to scarce terms, i.e. terms which do not appear in many documents. We calculate the IDF value for each term off-line by analysing a large corpus of a general nature. Currently we use an 18 million word corpus of English from CELEX (http://www.kun.nl/celex/), but future studies will use larger corpora such as the 100 million word British National Corpus (http://info.ox.ac.uk/bnc/). The TF.IDF values for the reference corpus are then formed into a vector in the space of all possible terms. A vector is similarly formed from the **title** of the candidate document, and the similarity of the two vectors, and hence the similarity of the reference corpus and the candidate title, is found by taking the cosine of the angle between them. In the field of information retrieval, M1 is typically used to compare a user query with the text of a candidate document. In our case, we are treating the entire reference corpus as a query, so the validity of this may require further investigation. However, we believe that the use of document titles (where they exist) is important in the current application, since authors give considerable thought to capturing the essence of a document in the title. Of course, this does not guarantee that the language inside is what one is looking for.

M2 takes the reference corpus and the entire candidate document, and for each, derives a set of character n-gram frequency lists, currently from n=2 to n=5. The two frequency distributions are then compared using the log likelihood statistical measure. The log-likelihood statistic (Dunning, 1993) does not suffer from some of the shortcomings of other metrics, e.g. chi-square, (which tends to over-emphasise high contingency values) and mutual information (which tends to over-emphasise low contingency values). In addition, it makes no assumption of normally distributed data and copes well with varying amounts of information, which is typically the case with data extracted from textual sources. The higher the resulting value, the more dissimilar are the two distributions. The advantage of M2 is that since any given text will contain more character n-grams than words, this method gives robust performance even on very short documents (where word-based techniques would suffer from data sparsity). In addition, it is known that a character n-gram distribution extracted from one subject area will differ significantly from that of another, while n-gram distributions from the same domain tend to share many common features. It is this property that enables character-level n-gram data to be used effectively for text categorisation (Cavnar & Trenkle, 1994).

M3 is the same as M2 except that instead of deriving a set of character n-gram frequency lists from the reference corpus and the candidate document, we derive simply a word frequency list from each. The two distributions are again compared using the log-likelihood measure. It is possible to omit the frequencies of very common words from this comparison.

M4 takes the reference corpus and builds a back-off n-gram language model (Katz, 1987) from it. Currently, a trigram model is built; higher n-grams could be used if warranted by the size of the reference corpus, but in its intended use the agent will generally be starting with a small seed corpus. The resulting language model is then "applied" to the candidate document to derive the perplexity of the latter, and this perplexity is taken as a dissimilarity measure between the reference corpus and the candidate document.

Let the results of applying the four metrics M1 to M4 to the candidate document be V1 to V4, respectively. Then the agent calculates an overall document "dissimilarity" measure DS, using the formula DS = W1 (1 - V1) + W2.V2 + W3.V3 + W4.V4, where W1 to W4 are weights for each metric, which determine how much it will contribute to the overall dissimilarity measure. The agent then compares DS with an existing similarity threshold ST (manually or automatically set), and if the candidate document is within the threshold (DS < ST), it is retained, ultimately for augmenting the reference corpus.

It should be noted that the agent will not necessarily return all suitable documents from a site due to the variability of Internet connections, time-outs, and so on. However, this is usually a minor consideration and language characteristics remain the prime determinant in the selection of documents for retention.

3 Experimental Work

Results will be presented for two experiments. Time constraints meant that these were in the nature of pilot studies. Much more extensive experiments are required to evaluate the algorithm thoroughly. In the first, we used metric M4 alone; in the second, we used metrics M2 to M4 (M1 was not used, since the development corpus, from which the similarity threshold was calculated, did not have a title). We discuss the perplexity of the resulting language models, and the behaviour of the four metrics in the overall algorithm.

3.1 Pilot Study 1

In this study, metric M4 alone was used. We set W4 = 1, which means that the dissimilarity measure calculated by the agent is given by DS = V4, where V4 is the result of applying M4 to the candidate document, i.e. the perplexity (PP) of the candidate document when the language model derived from the reference corpus is applied to it. The agent's success can be measured by the reduction in PP as measured against a test corpus, as more and more language data is returned. However, the calculation of a suitable dissimilarity threshold for PP must be made independently of the test data, so a third "development" corpus is needed. For this reason, the original source corpus was split into three sections: one to create the original LM for M4, one to determine the threshold, and one to test the final LM which incorporates the new data returned by the agent. The overall procedure is thus:

1. Choose a suitable source of training/test data. In this case the selected source was the contents of the BT Language Group web server.
2. Download the entire contents of this server.
3. Split the contents (randomly) into three equal sections: training, development and test.
4. Build a LM from the training data, using the CMU toolkit (Rosenfeld, 1994).
5. Measure the PP of the LM on the development corpus (PP_{dev}).
6. Direct the agent to a fixed list of relevant web servers (in this case, those linked to ELSNET's page entitled "*list of related servers*": http://www.elsnet.org/related-servers) with instructions to download documents with $PP_{doc} < PP_{dev}$, where PP_{doc} is the perplexity of the candidate document. In addition, the agent was also sent to a set of three local (but otherwise unrelated) BT servers.
7. When the agent has returned, measure the PP of the original LM on the test data.
8. Augment the training corpus with the newly found material.
9. Build a new LM therefrom, and measure PP on the test data.
10. Compare the two PP values (before and after augmentation).

3.2 Results

The contents of the BT Language web server produced a corpus of 93,459 words. This was divided into three subcorpora, as follows (showing the number of lines, words & characters):

1645	31495	188075	dev.txt
1644	30811	184031	test.txt
1645	31153	185202	train.txt

Although this source is not ideal, it shows apparent internal consistency, is well maintained and produces a corpus of a size not untypical of many speech recognition domain data sets. The PP of the LM created from train.txt against the development corpus (dev.txt) was 195.33, so this value was used as the threshold for similarity. When the agent was directed to the sites described above, their contents were analysed and a number of documents retained. The top three sites (in terms of quantity of text retained) are shown in Table 1. The quantity of text retained from the remaining sites was either negligible or zero.

The evaluation then proceeded by iteratively adding further text to the training corpus, rebuilding the LM, and calculating the PP on the test set. The construction of the LMs was performed using the CMU toolkit, which greatly simplifies the process and allows the creation of many types of statistical language model that are directly usable by a variety of recognition systems. It should be noted that the vocabulary used to construct the LM remained fixed throughout this study, since changes in vocabulary

WORDS	SITE	URL
36,292	BT Labs	http://www.labs.bt.com
16,805	CUED Speech FAQ	http://svr-www.eng.cam.ac.uk/comp.speech/
8,349	Johns Hopkins CLSP	http://www.clsp.jhu.edu/

Table 1: Quantity of text retained by the agent

Number	Corpus	Size (words)	PP(test)	change
1	train	31,153	206.06	0
2	train+BTLabs	67,445	240.77	+34.17
3	3+CUED	84,250	255.74	+49.14
4	4+CLSP	92,599	264.89	+58.83

Table 2: PP changes as the training corpus is augmented and the LM rebuilt.

size can have a significant effect on PP. However, vocabulary acquisition is clearly a further important facet of the language model and future studies will address this issue. The bigram and trigram cut-offs also remained static (both at zero). The results are shown in Table 2.

3.3 Discussion

Clearly, the PP has increased with the size of the training corpus. This was somewhat unexpected, as each individual document added to the corpus had a perplexity lower than that of the development set. As a comparison, therefore, the training corpus was augmented with the development corpus and a further LM built and tested. The result (in Table 3) shows that when the training corpus is augmented with additional material from the original source then the PP does indeed decrease. Clearly, further adjustment of the parameters associated with the similarity metrics and a review of the targeted web sites is needed if this method is to succeed in finding suitable training data. One simple change, when using M4 alone, might be to lower the similarity threshold, e.g. to say $PP_{doc} < PP_{dev}$ -20. Alternatively, the best way to improve performance may be to make use of the other metrics (e.g. word frequency & character n-grams). Consequently, the next evaluation was a repeat of the above experiment but using more than one metric.

3.4 Pilot Study 2

Experiment 2 proceeded as Experiment 1 but this time all the metrics except M1 were applied. The weights W2 to W4 were set at:

W2 = 0.1, W3 = 1.0, W4 = 10.0

since these weights had been found empirically to produce roughly equivalent contributions from each metric when averaged over a number of trials. However, in future trials it is hoped to use a more sophisticated method for setting the weights (see Section 4).

3.5 Results

The agent was then directed to the three sites that returned the most material in Experiment 1, returning all documents with a value of DS below the threshold, when the three metrics M2 to M4 were applied. The quantity of text returned is shown in Table 4.

Interestingly, the quantity of text returned from each site has changed radically. The BT site returns over ten times as much material, as the agent has retained the vast majority of the pages it analysed. This suggests that the similarity score threshold was far too loose. Similarly, the CLSP site now returns over four times as much material. Conversely, the result from the CUED site appears to be at odds with the others: it now returns just over half the previous quantity.

Once the material had been downloaded, the evaluation then proceeded as before: iteratively adding further text to the training corpus, rebuilding the LM, and calculating the PP on the test set. Again, the vocabulary and n-gram cut-offs remained fixed. The training corpus was augmented incrementally, in the order shown in Table 4. The quantity of material added at each stage was controlled to be consistent with Pilot Study 1 (although since the CUED server returned less material this time, more had to be added from CLSP to produce an equivalent final size). The results are shown in Table 5.

3.6 Discussion

As with Experiment 1, the PP increases with the size of the training corpus. However, this time, the rate of increase is far slower (approximately half of the previous value). This suggests that although a far greater quantity of material has been returned overall, its quality (for LM construction) is better. We assume that this is due in part to the use of

No.	Corpus	Size (words)	PP(test)
1.	train	31,153	206.06
2.	train+dev	62,648	150.71

Table 3: PP changes as the training corpus is augmented and the LM rebuilt.

WORDS	SITE	URL
492,032	BT Labs	http://www.labs.bt.com
68,816	Johns Hopkins CLSP	http://www.clsp.jhu.edu/
9,169	CUED Speech FAQ	http://svr-www.eng.cam.ac.uk/comp.speech/

Table 4: Quantity of text returned by the agent

metrics M2 and M3 as well as M4, although the limited nature of our experiments does not allow us to disentangle the effect of the particular weight settings chosen.

Since the figures in Table 5 suggest that the rate of PP increase is slowing, it is of interest to determine the extent to which this effect would continue. To investigate this, a further cycle of model building and testing was initiated, using *all* of the text returned by the agent. The material was added in the same order as before (i.e. BT then CUED then CLSP) in chunks of 100K words wherever possible (evidently since CUED and CLSP together returned less than 100K words their entire supply had to be used in a single chunk). The results are shown in Table 6.

This reveals an interesting pattern, as the PP increases and then seems to reach a plateau. The comparison between line (2) in Table 6 and line (4) in Table 5 is revealing: although train+100K is a bigger corpus, it produces a lower PP than 3+CLSP (217.92 vs. 235.11). It is possible that the BT material is more similar than CUED and CLSP and therefore its addition, even in larger quantities, has a less adverse effect on the PP.

In conclusion, the pilot experiments were successful, in that the agent has brought back material which is clearly similar to the seed corpus, and that our LM is now derived from a much larger corpus and in that sense more statistically reliable. However, our use of the agent has not yet led to a new LM with reduced test set perplexity. It might be argued that if the test set (which is simply one third of the original corpus) is too small, then test set perplexity is not in fact a reliable metric. (It should be borne in mind that the ultimate test for speech applications is recogniser performance on unseen utterances once the application is up and running). But for the meantime, test set perplexity is our benchmark, and we need both a deeper theoretical analysis of what factors are expected to cause it to rise or fall when the training material is augmented by the agent, and a much more thorough investigation of the contribution of each metric and the effect of the weight settings.

4 Further Work

4.1 Initial Setting of the Weights

In the studies described above, the weights for each metric were set manually, using settings that had been empirically shown to produce values of roughly equal magnitude from each of the metrics. However, it is possible (and indeed desirable) to set these weights automatically, using characteristics of the seed corpus. One important characteristic is the quantity of text; another is its *homogeneity.* If the seed corpus contains a multiplicity of authors and document types there will often be considerable internal variation, and the corpus may not constitute a coherent information source.

We propose that the initial weight values may be determined by a confidence value C, which is derived from the word count WC and the homogeneity H of the reference corpus. H is a measure of how internally self-consistent a text is. H is calculated by randomly allocating sentences

No.	Corpus	Size (words)	PP(test)	change
1.	train	31,153	206.06	0
2.	train+BTLabs	67,385	223.80	+17.74
3.	2+CUED	76,554	226.17	+20.11
4.	3+CLSP	92,620	235.11	+29.05

Table 5: PP changes as the training corpus is augmented and the LM rebuilt.

No.	Corpus	Size (words)	PP(test)	change
1.	train	31,153	206.06	0
2.	train+100K	131,097	217.92	+11.86
3.	train+200K	231,894	236.68	+30.62
4.	train+300K	331,651	263.28	+57.22
5.	train+400K	431,451	262.42	+56.36
6.	train+500K	532,354	289.73	+83.67
7.	train+570K	601,170	263.34	+57.28

Table 6: PP changes as the training corpus is augmented and the LM rebuilt.

from the reference text to one of two "sub-texts", and then comparing those two sub-texts with each other, using M3 as described above. In broad terms, if WC is high, then one can derive a more reliable word n-gram language model, and so W4 should be relatively high, whereas if WC is low, one needs to rely more on M2, which works at the character level. Similarly, the greater the value of H, the more reliable are similarity measurements based on the reference corpus. Ideally, the values of WC and H should control the setting of the initial weights in quite a sophisticated way, but for the present, they may simply be combined using the formula C = WC*H, followed by a binary decision: if C is greater than a given threshold, the weights are set to W1=W2=W3=W4=1, while if C is below this threshold, the weights are set to W1=1, W2=10, W3=1, W4=0.1.

4.2 Weight Updating

In the current work, the weights remain fixed at their initial values, which is potentially a major drawback, since one or more "poor metrics" may degrade the overall action of the agent, causing it to retain a lot of unwanted material. Consequently, we present two possible ways of dynamically adjusting weights during the operation of the agent. The first is to combine the retained documents (i.e. those which are sufficiently similar) with the reference corpus, and recalculate the weights using the original initial weights formula. As the size of the corpus grows, W4 should grow relative to the other weights, reflecting increased confidence in the word n-gram statistics. However, this relies on a more sophisticated initial weight setting formula than we have currently suggested.

The second way of dynamically adjusting weights is to iteratively adjust them on the basis of each retained document brought back by the agent. Each retained document is separately combined with the reference corpus, and used to calculate a new language model M4. M4 is then applied to a separate development corpus, which is normally a reserved part of the original reference corpus, and a perplexity value PP(new) is derived. This perplexity is compared with the original perplexity of the reference corpus language model applied to the development corpus (PP(old)). If a document is good for the purpose of building a language model, then the development set perplexity should decrease (or at least not increase by very much) when this document is added to the training corpus. One can then look at the metrics which lead to the retention of a "good" document, and increase their weights in proportion to their contribution to the overall DS value. This leads to the weight adjustment formula: $W_i = W_i + k. (W_i.V_i/DS).(PP(old) - PP(new)).W_i$, where i is an index running over each of the four weights. Note that this formula allows for both the incrementing and decrementing of weights, and that an individual weight may become negative, meaning that there is then a negative correlation between that metric and the similarity of a candidate document to the reference corpus.

5 Conclusion

A new agent has been described, which augments a seed corpus with similar data found on the Internet for the purpose of language model construction. The agent makes use of a combination of similarity metrics operating at a number of textual levels. Preliminary experiments have been inconclusive as to the future success of our method in producing better language models, but a potentially powerful algorithm has been described, with many possibilities for further investigations.

References

(Cavnar & Trenkle, 1994) W. Cavnar & J. Trenkle “N-Gram-based Text categorisation”, *Proceedings of the Symposium on Document Analysis and Information Retrieval,* Las Vegas, NV. (1994)

(Dunning, 1993) E. Dunning “Accurate methods for the statistics of surprise and coincidence”, *Computational Linguistics,* Vol. 19, No. 1. (1993)

(Katz, 1987) S. Katz “Estimation of probabilities from sparse data”, *IEE Trans. on Acoustics, Speech & Signal Processing*, vol. ASSP-35. (1987)

(Rose & Wyard, 1997) T. Rose & P.J. Wyard "A Similarity-Based Agent for Internet Searching", *Computer-Assisted Searching on the Internet (RIAO '97),* Montreal, Canada. (1997)

(Rosenfeld 1994) R. Rosenfeld “The CMU Statistical Language Modelling Toolkit, ARPA SLT ‘95. (1994)

(Rosenfeld 1996) R. Rosenfeld "A maximum entropy approach to adaptive statistical language modelling", Computer Speech and Language, 10, 187-228. (1996)

(Rudnicky 1995) A. Rudnicky "Language Modelling with limited domain data", Proc. ARPA Workshop on SLT, Morgan Kaufmann, San Mateo, pp. 66-69. (1995)

(Salton & McGill 1983) G. Salton, J. McGill “*Introduction to Modern Information Retrieval*”, McGraw Hill. (1983)

(Vergyri (1995) D. Vergyri "Exploiting Remote Domains via Data Bleaching", http://www.clsp.jhu.edu/~dverg/bleaching.html (1995)