The custom decay language model for long range dependencies

Mittul Singh[1,2], Clayton Greenberg[1,2,3], and Dietrich Klakow[1,2,3] ⋆

[1] Spoken Language Systems (LSV)
[2] Saarbrücken Graduate School of Computer Science, Saarland Informatics Campus
[3] Collaborative Research Center on Information Density and Linguistic Encoding
Saarland University, Saarbrücken, Germany
`{firstname.lastname}@lsv.uni-saarland.de`

Abstract. Significant correlations between words can be observed over long distances, but contemporary language models like N-grams, Skip grams, and recurrent neural network language models (RNNLMs) require a large number of parameters to capture these dependencies, if the models can do so at all. In this paper, we propose the Custom Decay Language Model (CDLM), which captures long range correlations while maintaining sub-linear increase in parameters with vocabulary size. This model has a robust and stable training procedure (unlike RNNLMs), a more powerful modeling scheme than the Skip models, and a customizable representation. In perplexity experiments, CDLMs outperform the Skip models using fewer number of parameters. A CDLM also nominally outperformed a similar-sized RNNLM, meaning that it learned as much as the RNNLM but without recurrence.

Key words: Reduction in number of parameters, robust training, long range context, language model

1 Introduction

The central task of automatic speech recognition (ASR) is predicting the next word given sequential acoustic data. Language models (LMs), which predict words given some notion of context, inform ASR systems about which word choices fit well together, thus acting complementarily to acoustic models which directly assign probabilities to words given the acoustic input. Within ASR Systems, smoothed N-gram LMs have been very successful and also are very simple to build. These standard LMs work well on short context sizes because the model directly enumerates them. But enumerating dependencies of longer distances is unfeasible due to the exponential growth of parameters it would require.

To quatify information in dependencies of long distances, we use a variant of pointwise mutual information. Specifically, for a given pair of words (w_1, w_2)

⋆ The work was supported by the Cluster of Excellence for Multimodal Computing and Interaction, the German Research Foundation (DFG) as part of SFB 1102 and the EU FP7 Metalogue project (grant agreement number: 611073).

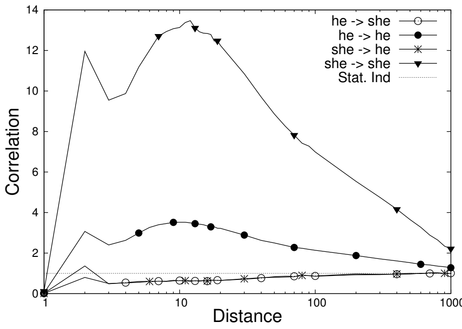

Fig. 1. Variation of word triggering correlations for pronouns over large distances

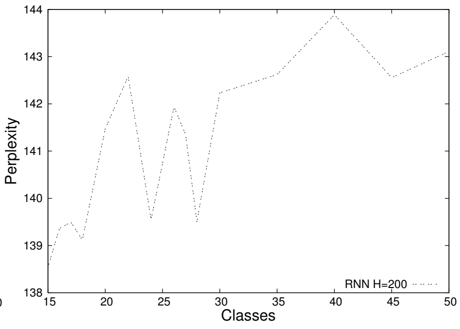

Fig. 2. Variation of perplexity against the number of classes for a RNNLM with 200 hidden nodes

separated over a distance d, we examine the ratio of the actual co-occurrence rate to the statistically predicted co-occurrence rate: $c_d(w_1, w_2) = \frac{P_d(w_1,w_2)}{P(w_1)P(w_2)}$. A value greater than 1 shows it is more likely that the word w_2 follows w_1 at a distance d than otherwise expected according to the unigram frequencies of the two words. In Fig. 1, we show an example variation of this correlation for pronouns with the distance d on the English Gigawords corpus [1].

In this corpus, seeing another "she" about twenty words after seeing a first "she" is more than 13 times more likely than seeing a "she" in general. A similar, but interestingly weaker, observation can be made for the word "he". Note also that "she" somewhat suppresses "he" and vice versa, and these cross-correlations, although negative, are still informative for a prediction system. In summary, Fig.1 demonstrates that plenty of word triggering information is spread out over long distance dependencies that is typically beyond the reach of N-gram LMs.

Several models, such as the cache-based LM [2], Skip models [3, 4], and recurrent neural network language models (RNNLMs) [5] have been proposed to capture triggering in large contexts, but they usually only handle auto-triggering and/or have too many parameters to scale with vocabulary size. In this paper, we develop a novel modelling scheme, the Custom Decay Language Model (CDLM), which is specifically built to capture long range dependencies while growth in number of parameters remains sub-linear in vocabulary size. CDLMs outperform large-context-size Skip models, which are not constrained this way. Additionally, CDLMs show a more robust variation of performance metric against the variation of meta-parameters than RNNLMs, and they allow us to study the sparseness of word representations over different context sizes.

In the rest of the paper, we first briefly describe Skip models and RNNLMs and their limitations in Section 2, leading up to the detailed description of our new modelling technique in Section 2.3. We then set up experiments to analyze performance of these models in Section 3. Section 4 gives a robustness analysis of our model in addition to perplexity results for comparing the performance of various LM types and finally, Section 5 gives some concluding remarks.

2 Language models

In this section, we first briefly describe and outline the numbers of parameters needed by Skip models and RNNLMs for handling long range dependencies. We then describe our novel CDLM which has been designed to overcome the limitations of skip models by reducing the number of parameters.

2.1 Skip models

Skip models enumerate dependencies like N-grams, but allow wildcards (skips) at specific positions. This technique in combination with distance-specific smoothing methods spans larger context sizes and reduces the sparseness problem. However, the number of parameters still grow by $O(V^2)$ (where V is the vocabulary size) each time the context size is increased by one, making them computationally inefficient. In addition, the skip modeling framework lacks representational power when compared to neural network based LMs.

For our experiments, we build skip models by combining unified-smoothing trigrams and distance bigrams, which extend the range. Previously, such a combination has been shown to outperform state-of-the-art smoothed N-gram LMs [6].

2.2 RNNLMs

RNNLMs provide impressive performance gains when compared to other state-of-the-art language models. Through recurrence, the context size for these models is essentially infinite, or at least, formally unconstrained. This makes them especially suitable for long range dependencies. However, training RNNLMs can be slow, especially because the output must be normalized for each word in the vocabulary. Hierarchical softmax and related procedures that involve decomposing the output layer into classes can help with this normalization [7]. Unfortunately, using classes for normalization complicates the training process, since it creates a particularly volatile metaparameter. This can be observed in Fig. 2, where even for small variation in classes, RNNLMs show unstable variation in perplexity.

In our experiments, we employ a widely used class-based RNNLM implementation [5] builds networks that require $H^2 + 2HV + HC$ parameters, where H is the number of hidden units and C is the number of normalization classes. To produce better RNNLMs, we can increase the hidden layer size by one which in turn increases the number of parameters linearly in vocabulary size ($O(V + H + C)$).

2.3 Custom decay language models

Our new modelling scheme was inspired by log-linear language models, which are characterized by sub-linear growth in the number of parameters with context size [8]. This model consists of two parts: a log-linear model and an N-gram model. For a history of size M, the N-gram part looks at the first $N - 1$

$(N < M)$ predecessor words and the log-linear part captures the triggering information stored in distances d in the range $[N, M)$. Given the string of words $\{w_{i-M+2}, \cdots, w_{i-1}, w_i, w_{i+1}\}$ where $h = \{w_{i-M+2}, \cdots, w_i\}$, and supposing that $N = 3$, CDLM can be defined as :

$$\begin{aligned} P(w_{i+1}|h_i) = \ & \tfrac{1}{Z(h_i)} \times P_{3\text{-}gram}(w_{i+1}|w_{i-1}, w_i) \\ & \times e^{(E^{w_{i+1}} v_{w_{i-2}} + \sum_{k=i-N+2}^{i-3} E^{w_{i+1}} T_k v_{w_k})} \end{aligned} \tag{1}$$

where i is the position in the document, $P_{3\text{-}gram}$ is a standard trigram LM and v_{w_k} is the vector representation of the word at a distance k from the word to be predicted in a C-dimensional, continuous, dense latent space $(C < V)$. Here, the dimensions of C can be understood as “classes” capturing latent semantic information in the data.

E^{w_i} refers to a column of the emission matrix E, which weighs the word vectors v_{w_k} to predict the next word. Such a matrix can be thought of as an interpretation function for the current latent state of the model. These latent states exist in the same space as the word vectors. Presumably, some words are closer to this state than others. In this way, the latent states represent semantic concepts that the E matrix can translate into words.

The model also includes a distance specific transition matrix T_k to take word vectors from one distance-based latent space to another. More directly, the T_k matrices control the decay of each word within the latent state. Since the T_k are matrices, as opposed to scalars, which would provide a uniform decay, and as opposed to vectors which would provide a class-based decay, the shape of the decay function is *custom* to each word, which is why this model is named the Custom Decay Language Model.

This setup allows the model to constrain the number of parameters, as each time a word is added to the latent state, only the T_k matrix needs to be updated. Apart from the $O(V^3)$ parameters required to construct the trigram, it needs $O(VC)$ parameters to train the E matrix and the word vectors v_{w_k}, and it needs $O(C^2)$ parameters for training the T_k matrices. In all, CDLM parameters increase sub-linearly with V.

As shown in the last line of Equation 1, the model log-linearly combines $T_k v_{w_k}$ at each context position to form a long-distance predictor of the next word. This approach, though inspired by skip models, is more customizable as it allows the exponent parameters to include matrix based formulations and not be constrained only to single values like skip models. Though the exponential element captures the latent/topical information well, the effects are too subtle to capture many simple short-distance dependencies (sparse sequential details). In order to make the model richer in sparse sequential details, we log-linearly combine the long-distance component with an N-gram LM.

In order to estimate the parameters E, v_{w_k} and T_k, we use the stochastic gradient descent algorithm and minimize the training perplexity of CDLM.

3 Language modeling experiments

3.1 Corpus

We trained and evaluated the LMs on the Penn Treebank as preprocessed in [9]. We used the traditional divisions of the corpus: sections 0-20 for training (925K tokens), sections 21-22 for parameter setting (development: 73K tokens), and sections 23-24 for testing (82K tokens). Despite its final vocabulary of 9,997 words and overall small size, this particular version has become a standard for evaluating perplexities of novel language models [10, 11]. The small size makes training and testing faster, but also makes demonstrating differences in performance more difficult. We expect our results would scale for larger datasets.

3.2 Experimental Setup

In our experiments, we use perplexity as the performance metric to compare the language modelling techniques described in this paper.

In order to establish the most competitive baselines, the RNNLMs trained in our experiment were optimized for number of classes. Recall that these classes just aid the normalization process, as opposed to CDLM classes, which form a very integral part of the model. If classes were overhauled from the RNNLM altogether, training would take much longer, but the perplexity results would be slightly lower. We found that 15 classes optimized perplexity values for RNNLMs with 50 and 145 hidden nodes, and 18 classes optimized perplexity values for RNNLMs with 500 nodes. These models were trained using the freely available RNNLM toolkit, version 0.4b, with the `-rand-seed 1` and `-bptt 3` arguments.

The N-gram models used were trained with SRILM. They were a unified smoothing trigram (*UniSt*) and an interpolated modified Kneser-Ney 5-gram (*KN*). The *KN* model was trained with the following arguments: `-order 5 -gt2min 1 -gt4min 1 -gt3min 1 -kndiscount -interpolate`.

CDLM uses the unified-smoothing trigram as the short-distance dependency component of its model and the long-distance (exponential) element of the model considers up to five words after the trigram.

The learning rate (η) adaptation scheme is managed by the adaptive gradient methods [12]. After optimizing on the development set, η was fixed to 0.1 and the dimensionality of the latent space C was fixed at 45.

While building CDLMs, we first trained a CDLM $M = 4$ and reused its constituent parameters E and v to build CDLM $M = 5$, only updating T_k while training. This process iterated up to $M = 8$.

4 Results and Discussion

4.1 CDLM Robustness Analysis

CDLM shows robust variation of perplexity with changes in classes, as shown in Fig. 3. The perplexity values decrease monotonically with increasing classes, as expected since each increase in class creates more parameters that can be tuned. Note that moving from $M = 4$ to $M = 5$ doubles the number of T_k matrices,

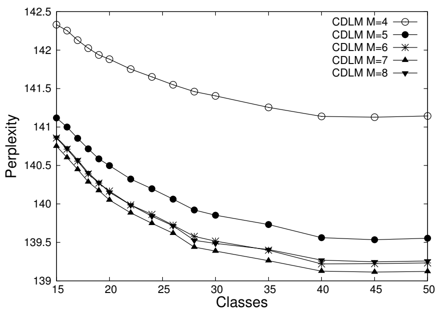

Fig. 3. Perplexity versus number of classes (C) in CDLM

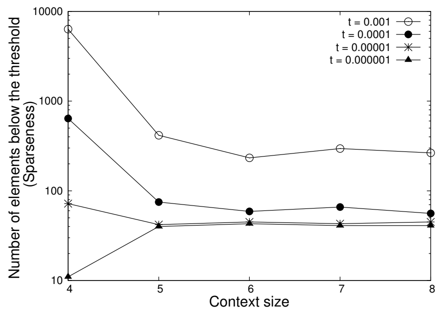

Fig. 4. Sparseness of CDLM's transformed word space $(T_l v_{w_l})$ measured at different threshold (t) versus its context size

which caused the large perplexity drop.

Along with the robustness shown by CDLM, the log-linear formulation of CDLM allows us to study and analyze the sparseness of the transformed word space matrices represented by $T_k v_{w_k}$ for different distances. We measure sparseness by counting the matrix entries below a given threshold. By this measure, a more sparse matrix will have large number of entries below the threshold than a less sparse matrix. We plot the variation of the sparseness for $T_k v_{w_k}$ matrices for different thresholds against the context size of CDLM in Fig.4. In most cases, we observe that as the context size increases the transition matrices have fewer number of entries below the threshold making them less sparse. Therefore, we believe that this matrix formulation alleviates the sparseness problem and also allows the exponent part to capture latent information.

4.2 Perplexity results

Table 1 presents our comparison of CDLM with different language models on the basis of their total numbers of parameters and their perplexities. As shown, skip models (*Skip*) outperform the unified smoothing trigram (*UniSt3*) as they have more parameters and hence, they better encode information spread over larger distances.

CDLM outperforms *UniSt3* because of spanning larger context size and greater number of parameters at its disposal. *CDLM45* also outperforms the *Skip* models. In fact, increasing the context size of *Skip* to eight words obtains a perplexity of 153.2, which is still less than the CDLM perplexity of 141.1 for a context size of four words. Also, *Skip* requires 4.1 million parameters which is more than a third greater than those required to build the seven-word CDLM. Also, *CDLM* is able to perform better than *KN* with fewer number of parameters. When combining *CDLM* with *KN*, increasing the context size for CDLM obtains progressively-better performance than *KN*. This is due to more number of parameters in CDLM formulation.

We further compare CDLMs with RNNLMs. An RNNLM with 145 hidden nodes has about the same number of parameters as *CDLM* and performs 0.1

Table 1. Test set perplexity (PPL) and total number of parameters (PAR) for each language model (LM).

LM	Range	Hidden	PPL	PAR
UniSt	3	-	162.1	2.0M
	4		160.0	
	5		155.8	
Skip	6	-	154.4	4.1M
	7		153.6	
	8		153.2	
KN	5	-	141.8	3.2M
		50	156.5	1.0M
RNNLM	∞	145	139.3	2.9M
		500	136.6	10.3M
	4		141.1	
	5		139.5	
CDLM	6	45	139.2	2.9M
	7		139.1	
	8		139.2	
	$5+4$		137.2	
	$5+5$		135.7	
KN+CDLM	$5+6$	45	135.2	6.1M
	$5+7$		134.9	
	$5+8$		134.9	
KN+RNNLM	$5+\infty$	50	120.3	4.2M
CDLM+RNNLM	$7+\infty$	$45+50$	120.2	3.9M

perplexity points worse than *CDLM.* Increasing the hidden units for RNNLM to 500, we obtain the best performing RNNLM. This comes at a cost of using a lot of parameters. To produce better performing LMs with fewer parameters we constructed an RNNLM with 50 hidden units, which when linearly combined with *CDLM* (*CDLM+RNNLM*) outperforms the best *RNNLM* using less than half as many parameters. It even nominally outperforms the combination of *KN* and *RNNLM* using fewer parameters, but this difference is likely not significant. Combinations of the three different LMs do not result in any large improvements, suggesting that there is redundancy in the information spread over these three types of LMs.

Finally, we note that the increase in parameters does not always lead to better performance. We observe this increase while comparing a *Skip* model with *CDLM* and this increase can be attributed to the richer formulation of *CDLM.* Increase in parameters for *CDLM+KN* does not also lead to a better performance against the fewer-parameters *CDLM+RNNLM.* This is also observed when we compare, *CDLM+KN* and *KN+RNNLM.* In this case, we suspect that the lower performance is due to CDLM's lack of recursive connections which form an integral part of RNNLMs. But note that CDLMs, which are not recurrent, can capture much of the long-distance information that the recurrent language models can.

5 Conclusion

In this paper, we proposed Custom Decay Language Model, inspired by Skip models' log-linear technique of dividing context into smaller bigrams and then recombining them. In contrast with Skip models, CDLM uses a richer formulation by employing a matrix based exponentiation method to capture long range dependencies. Additionally, CDLM model uses an N-gram model to capture the short range regularities.

Perplexity improvements are observed for CDLM even when compared to Skip models with larger range and Kneser Ney five-grams. This improvement is observed even though CDLM uses fewer parameters compared to larger range Skip model and $KN5$ with more parameters. In conclusion, CDLM provides a rich formulation for language modeling where the growth of number of parameters is constrained. We look forward to further enhancing CDLM with recurrent connections and analyzing its performance on other language datasets with a focus on ASR tasks.

References

1. David Graff and Christopher Cieri, "English gigaword ldc2003t05," Web Download. Philadelphia: Linguistic Data Consortium, 2003.
2. Roland Kuhn and Renato De Mori, "A cache-based natural language model for speech recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 6, pp. 570–583, 1990.
3. David Guthrie, Ben Allison, Wei Liu, Louise Guthrie, and Yorick Wilks, "A closer look at skip-gram modelling," in *Proceedings of the 5th international Conference on Language Resources and Evaluation (LREC)*, 2006, pp. 1222–1225.
4. Saeedeh Momtazi, Friedrich Faubel, and Dietrich Klakow, "Within and across sentence boundary language model," in *11th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, Makuhari, Japan, September 2010, pp. 1800–1803.
5. Tomas Mikolov, Stefan Kombrink, Lukas Burget, Jan Cernocký, and Sanjeev Khudanpur, "Extensions of recurrent neural network language model," in *2011 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Prague, Czech Republic, May 2011, pp. 5528–5531.
6. M. Singh and D. Klakow, "Comparing RNNs and log-linear interpolation of improved skip-model on four babel languages: Cantonese, pashto, tagalog, turkish," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2013, pp. 8416–8420.
7. Joshua T. Goodman, "Classes for fast maximum entropy training," in *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Salt Lake City, Utah, USA, May 2001, IEEE, vol. 1, pp. 561–564.
8. Dietrich Klakow, "Log-linear interpolation of language models," in *Fifth International Conference on Spoken Language Processing (ICSLP)*, 1998, pp. 1695–1698.
9. Eugene Charniak, "Immediate-head parsing for language models," in *Proceedings of 39th Annual Meeting of the Association for Computational Linguistics*, Toulouse, France, July 2001, pp. 124–131.
10. Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernocký, and Sanjeev Khudanpur, "Recurrent neural network based language model," in *11th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, Makuhari, Japan, September 2010, pp. 1045–1048.
11. Wei-Chen Cheng, Stanley Kok, Hoai Vu Pham, Hai Leong Chieu, and Kian Ming A Chai, "Language modeling with sum-product networks," in *15th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, Singapore, September 2014, pp. 2098–2102.
12. John Duchi, Elad Hazan, and Yoram Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *The Journal of Machine Learning Research*, vol. 12, pp. 2121–2159, July 2011.