An infinite replicated Softmax model for topic modeling

Nikolas Alexander Huhnstock[1(✉)], Alexander Karlsson[1], Maria Riveiro[1,2], and H. Joe Steinhauer[1]

[1] School of Informatics, University of Skövde, Skövde, Sweden
{nikolas.huhnstock,alexander.karlsson,joe.steinhauer}@his.se
[2] School of Engineering, Jönköping University, Jönköping, Sweden
maria.riveiro@ju.se

Abstract. In this paper, we describe the infinite replicated Softmax model (iRSM) as an adaptive topic model, utilizing the combination of the infinite restricted Boltzmann machine (iRBM) and the replicated Softmax model (RSM). In our approach, the iRBM extends the RBM by enabling its hidden layer to adapt to the data at hand, while the RSM allows for modeling low-dimensional latent semantic representation from a corpus. The combination of the two results is a method that is able to self-adapt to the number of topics within the document corpus and hence, renders manual identification of the correct number of topics superfluous. We propose a hybrid training approach to effectively improve the performance of the iRSM. An empirical evaluation is performed on a standard data set and the results are compared to the results of a baseline topic model. The results show that the iRSM adapts its hidden layer size to the data and when trained in the proposed hybrid manner outperforms the base RSM model.

Keywords: Restricted Boltzmann machine · Unsupervised learning · Topic modeling · Adaptive Neural Network

1 Introduction

One important task of data analysis is clustering which usually addresses the problem of grouping data points that share similar conceptual characteristics into groups. Analyzing textual data can be seen as a special case of clustering. Here, clusters, often referred to as topics, consist of words that are frequently occurring together.

Topic modeling algorithms are statistical methods for analyzing co-occurrences of words within text documents to discover topics that can be further used for categorization purposes within different application scenarios [1]. One of the most widely employed topic model is the latent Dirichlet allocation (LDA) [10], introduced by Blei and Jordan (2003) which is able to discover the thematic structure within large archives of text [1]. Each document within such a document corpus, as explained in [10, pp.5-6], can be regarded as a bag-of-words

that has been produced by the mixture of topics that the document's author intended to discuss. Each topic is hence represented by a distribution over all words that can be found in the document corpus. Abstractly speaking, when a document is generated, the author would repeatedly pick a topic, then a word belonging to that topic and places it in the bag until a document is complete. The objective of topic modeling is then to find the statistical parameters of such a process that is likely to have generated the corpus [10, pp.5-6].

Topic modeling algorithms work unsupervised and do not usually require any prior annotations or labeling of the documents since topics emerge from the original texts under analysis [1]. However, most topic modeling methods rely on manually setting important initial input parameters, such as the number of topics that is to be expected to be found in the document corpus [3]. The estimation of this rather crucial parameter is challenging and usually requires a certain level of knowledge about the content of document corpus that sometimes could be provided by human experts.

In this work, we attempt to overcome the challenge of determining the number of topics manually and propose a neural network-based approach to topic modeling that is able to self adapt the number of topics within a corpus of text. This method utilizes the combination of two recently developed extensions to the restricted Boltzmann machine (RBM) [13]: the replicated Softmax model (RSM) [6] which adapts the RBM to be usable for topic modeling, and the infinite restricted Boltzmann Machine (iRBM) [2] which is an adaptation of the RBM able to self identify the number of clusters needed for a traditional clustering problem. We combine these two different extension of the RBM into the infinite replicated Softmax model (iRSM) that is capable to self identify the number of topics within a corpus of text documents.

The remainder of the paper is structured as follows: In order to describe the aforementioned approach to topic modeling, we first provide some formal information and preliminaries regarding RBM, RSM and iRBM in section 2. This is followed by the presentation of relevant related work in section 3 after which the proposed model is introduced in section 4. Section 6 describes the empirical and qualitative evaluations of our approach and our results are presented in section 7. We conclude the paper with a brief summary of our findings and conclusions drawn, in section 9.

2 Preliminaries

Two main problems arise when trying to model the contents (represented by topics) of a corpus of textual documents with a RBM. Firstly, the number of words within documents may vary from one document to another and secondly, to infer topics from documents the number of topics that the model is able to represent has to be set in advance which requires knowledge about the corpus' contents which a user cannot be guaranteed to have.

In this section, the two methods the iRSM is based on are introduced: the RBM and the two different adaptions to it, namely the RSM and the iRBM.

2.1 RBM

The RBM [13] can be described as an undirected bipartite graphical model composed of one visible layer $\mathbf{v}$ and one hidden layer $\mathbf{h}$. A weight W_{ij} is associated with each connection between units v_i and h_j of the two layers. Given a binary RBM with n visible and m hidden units we can describe the energy of the model for a given state $(\mathbf{v}, \mathbf{h})$ as:

$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{c}^T\mathbf{h} - \mathbf{h}^T\mathbf{W}\mathbf{v} - \mathbf{b}^T\mathbf{v} \tag{1}$$

Due to its bipartite structure, states of visible and hidden units are only dependent on the other layers' units. The conditional distributions of the layers are therefore described by:

$$p(h_k = 1 \mid \mathbf{v}) = \sigma\left(c_k + \sum_{i=1}^{n} W_{ki} v_i\right), \tag{2}$$

$$p(v_k = 1 \mid \mathbf{h}) = \sigma\left(\sum_{j=1}^{m} h_j W_{jk} + b_k\right), \tag{3}$$

where $\sigma(x) = (1 + \exp(-x))^{-1}$.

2.2 RSM

The replicated Softmax model (RSM), proposed by Salakhutdinov and Hinton [6], has been used to enable the RBM to model documents of words. The RSM addresses the problem of a varying number of words within documents by allocating one visible unit per word in the document while sharing parameters (weights) over all visible units. Hence, it allows the RSM to model arbitrarily sized documents while decoupling the number of free parameters from the document length. This comes, however, at the cost of disregarding the order in which the words occur within the document.

When deploying an RSM, a document is modeled as binary matrix $\mathbf{U} \in \{0, 1\}^{V,D}$, where V is the number of words in the dictionary and D is the number of words in the document. The matrix $\mathbf{U}$ defines the observed state of the visible units $\mathbf{v}$ such that $U_k^\eta = 1$ is equal to the kth unit taking value η ($v_k = \eta$). The energy of the RSM given a state $(\mathbf{v}, \mathbf{h})$ is described by:

$$E(\mathbf{v}, \mathbf{h}) = -D\mathbf{c}^T\mathbf{h} - \sum_{i=1}^{n=D} \mathbf{h}^T\mathbf{W}_{\bullet, v_i} - \sum_{i=1}^{n} b_{v_i} \tag{4}$$

To balance the offset that has been introduced through the varying number of visible units that are contributing to the model's energy, the hidden bias term $\mathbf{c}^T\mathbf{h}$ is scaled according to the document's length D.

Since the bipartite structure of the RBM is preserved, the conditional distributions of hidden and visible units are given by:

$$p(h_k = 1 \mid \mathbf{v}) = \sigma\left(Dc_k + \sum_{i=1}^{n=D} W_{k,v_i}\right), \tag{5}$$

$$p(v_k = v^* \mid \mathbf{h}) = \frac{\exp\left(\sum_{j=1}^{m} h_j W_{j,v^*} + b_{v^*}\right)}{\sum_{t=1}^{V} \exp\left(\sum_{j=1}^{m} h_j W_{j,i}^t + \sum_{i=1}^{n} b_i^t\right)}. \tag{6}$$

2.3 iRBM

The infinite restricted Boltzmann machine [2] extends the RBM by enabling it to adapt the size of its hidden layer. This behavior is achieved by introducing a, in theory infinitely large, hidden layer $\mathbf{h}$ of that only a subset $\{h_j \mid j \leq z\}$ is considered. The number of hidden units describing this subset is given by the value of the introduced random variable z. The weights and biases associated with the hidden units $\{h_j \mid j > z\}$ are assumed to have a value of 0 and the energy of a given binary iRBM is given by:

$$E(\mathbf{v}, \mathbf{h}, z) = -\sum_{j=1}^{z} (c_j h_j - \beta_j) - \sum_{j=1}^{z} h_j \mathbf{W}_{j,\cdot}\mathbf{v} - \mathbf{b}^T\mathbf{v}. \tag{7}$$

To counteract the growth of z, Salakhutdinov and Hinton [2] introduced a penalty term β_j, which penalizes the accumulation of untrained units. The penalty term is parametrized on each hidden unit's bias with a global penalty β as $\beta_j = \beta\,\mathrm{soft}_+(c_j)$. With $\mathrm{soft}_+(x) = ln(1 + \exp(x))$

With the introduced random variable z the conditional distributions of the model are given by:

$$p(h_k = 1 \mid \mathbf{v}) = \begin{cases} \sigma(c_k + \mathbf{W}_{j,\cdot}\mathbf{v}), & k \leq z \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

$$p(v_k = 1 \mid \mathbf{h}) = \begin{cases} \sigma\left(\sum_{j=1}^{m} h_j W_{jk} + b_k\right), & k \leq z \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

$$p(z \mid \mathbf{v}) = \frac{\exp -F(\mathbf{v}, z)}{\sum_{z^*}^{\infty} \exp -F(\mathbf{v}, z^*)} \tag{10}$$

It can be shown that the infinite sum occurring in the denominator of (10) can be reformulated into a sum over a term of trained hidden units and a finite geometric series that can be a computed analytically, given that β is greater than 1 [2].

3 Related Work

Inspired by the RSM's weight sharing technique, Larochelle and Laury [8] extended the neural autoregressive distribution estimator (NADE) [9] and enabled

the model to represent documents. The so-called DocNade inherits the advantageous characteristic of computing the gradient of the negative log-likelihood over the data without requiring approximation. The DocNode uses a hierarchy of binary logistic regressions to represent the distribution of words, which results in a sublinear scaling with V when sampling the probability of an observed word. Although the DocNade architecture corresponds to several parallel hidden layers, i.e. one for each input word, with tied weights the number of units in each layer needs to be defined manually and is static.

Based on the RSM, Srivasta et al. [14] developed the Over-Replicated Softmax model, which belongs to the family of Deep Boltzmann Machines, i.e. Boltzmann Machines that contain at least two hidden layers. The Over-Replicated Softmax has softmax visible units and binary hidden units in the first layer and on top of that another softmax hidden layer. This is supposed to provide a more flexible prior over the hidden representations.

Srivasta et al. introduced this second hidden layer without the usual increase in model parameters, by reusing the weights that connect the visible layer to the first hidden layer, for the connections between the first and second hidden layer. This allows the Over-Replicated Softmax model to be trained as efficiently as the RSM despite the presence of an additional layer.

Even though the Over-Replicated Softmax model and the DocNade model achieved better results than the RSM model, both models require manual setting of the hidden layer(s), which is the fundamental issue that will be resolved within the proposed iRSM.

4 Proposed Model

The proposed model is a combination of the RSM [6] and the iRBM [2] which we refer to as the infinite replicated Softmax model (iRSM). It combines the capability of the RSM as an undirected topic model while, at the same time, it adapts to the number of represented topics automatically.

The iRSM can be trained on documents of varying length due to the use of the RSM's weight sharing technique, allowing it to replicate input units depending on the input document's length. Furthermore, the iRBM's hidden layer's growing behavior has been adopted by introducing a theoretical infinite hidden layer together with a growing penalty. Figure 1 shows a graphical illustration of the proposed model.

The energy function of the iRSM takes the following from:

$$E(\mathbf{v}, \mathbf{h}, z) = -D \sum_{j=1}^{z} (c_j h_j - \beta_j) - \sum_{j=1}^{z} \sum_{i=1}^{n=D} h_j W_{j,v_i} - \sum_{i=1}^{n} b_{v_i}, \qquad (11)$$

with $\beta_j = \beta \operatorname{soft}_+(c_j)$. The growing penalty β_j, defined in the same way as for the iRBM, enables the model to adapt its hidden layer size according to the inputs. In addition to the scaling of the hidden term of the RSM, the growing

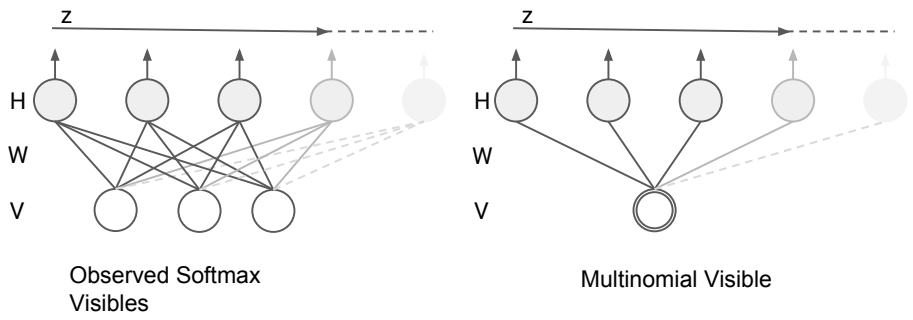


Fig. 1: Graphical representation of the iRSM. (left) An iRSM with three visible softmax units. (right) Visible softmax units replaced with a single multinomial unit which is sampled D times. The shaded hidden units indicate that these are added based on the state of z.

penalty β is scaled by the size of the document in order to maintain balance among terms.

Given an iRSM with binary hidden units, the conditional distributions are given by:

$$p(h_k = 1 \mid \mathbf{v}) = \begin{cases} D(c_k - \beta_k) + \sum_{i=1} n\mathbf{W}_{j,v_i}, & k \leq z \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

$$p(v_k = 1 \mid \mathbf{h}) = \begin{cases} \frac{\exp\left(\sum_{j=1}^{m} h_j W_{j,v^*} + b_{v^*}\right)}{\sum_{t=1}^{V} \exp\left(\sum_{j=1}^{m} h_j W_{j,i}^{t} + \sum_{i=1}^{n} b_i^t\right)}, & k \leq z \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

$$p(z \mid \mathbf{v}) = \frac{\exp -F(\mathbf{v}, z)}{\sum_{z^*}^{\infty} \exp -F(\mathbf{v}, z^*)} \tag{14}$$

The iRSM's learning parameters are obtained through the application of gradient descent on the model's negative log-likelihood (NLL) over documents. For a single document v this takes the form:

$$\frac{\partial - log(p((v)))}{\partial \theta} = \mathbb{E}_{\mathbf{h},z|v}\left[\frac{\partial}{\partial \theta} E(v, \mathbf{h}, z)\right] - \mathbb{E}_{\mathbf{v},\mathbf{h},z}\left[\frac{\partial}{\partial \theta} E(\mathbf{v}, \mathbf{h}, z)\right] \tag{15}$$

The computation of the second expectation term is considered to be too expensive since it involves the sum over all possible states of the network. Instead, Contrastive Divergence (CD) [5] can be used to approximate the gradient of the NLL by running a short Markov Chain wherein sampling is alternated between $z \sim p(z \mid \mathbf{v})$, $\mathbf{h} \sim p(\mathbf{h} \mid \mathbf{v}, z)$ and $\mathbf{v} \sim p(\mathbf{v} \mid \mathbf{h}, z)$.

5 Hybrid Training

Additionally to the iRSM model introduced previously we propose a hybrid training approach. The idea of this training method is to combine the iRSM and RSM into a two phase training procedure, where the former determines the networks hidden layer size and the latter is used to improve performance. The motivation behind this procedure is that the iRSM is, even in later stages of the training process, still slightly adjusting its hidden layer size. This behavior was as well observed in previous work for the iRBM in the context of clustering [7]. This leads to the situation, in which some of the already limited amount of information is continuously devoted to the task of adjusting the hidden layer's size. In order to leverage as much as possible from the sparse information, we decided to discard the adaptive behavior of the iRSM at a point in the training process where the iRSM has had sufficient time to develop its hidden layer. From this point forward the training is solely focused on optimising weight and bias parameter to learn the representation of the data as good as possible. By making the size of the hidden layer static at a given point in time, the parameters will be fine-tuned to that size of the hidden layer. The training process begins by training an iRSM; after some predefined time, e.g. half the total training time, this iRSM is transformed into a RSM and training continues until termination.

6 Experiment Design

In this section, we describe the empirical and qualitative experiments we conducted. Since Salakhutdinov and Hinton [6] showed that the base RSM is able to outperform Latent Dirichlet Allocation (LDA) we do not further go into the comparison with LDA and focus on comparing the RSM with iRSM and the iRSM trained in the proposed hybrid manner.

The first experiment quantitatively analyses the influence of the regularization parameter beta on the behavior of the iRSM and makes the comparison with the base RSM, we report mean and standard deviation of 10 trials. The second experiment provides an analysis of the top words per topic identified by an hybridly trained iRMS for different parameter settings. The "Reuters-21578, Distribution 1.0" corpus contains 10,788 news documents totaling 1.3 million words and was compiled by David Lewis[3]. The data set is split into 7,769 training documents and 3019 test documents. Common stopwords are removed from the data and the words are stemmed. To effectively reduce the dimensionality of the problem space, we only consider the 2000 most frequent words similar to what Salakhutdinov and Larochelle [6, 8] suggested by.

We use per word perplexity as a metric to assess the models generative performance through:

$$exp\left(-\frac{1}{N}\sum_{i=1}^{N}\frac{1}{D_i}\log p(\mathbf{v}_i)\right), \tag{16}$$

[3] Available at: http://www.daviddlewis.com/resources/testcollections/reuters21578/ [Accessed 22 May 2019].

where D_i represents the word count of the i-th document. The perplexity is evaluated in a similar fashion as Salakhutdinov [6] over 50 randomly held out test documents. Computing the probability of held-out documents exactly is intractable for undirected models, such as the RBM, since it requires to enumerate of over an exponential number of terms. Therefore, annealed importance sampling (AIS) [6] is deployed to obtain $p(\mathbf{v})$ of the RSM and iRSM by averaging over 100 runs using 1,000 in $[0, 1]$ uniformly spaced temperatures β.

To allow comparison between the models: all models processed the data in batches of size 100 and were trained for an equal amount of epochs. During training, the adaptive gradient algorithm ADAGRAD [4] with an initial learning rate of 0.05 is deployed.

7 Results

The results of the first experiment are depicted in Figure 2. The plot shows average perplexity scores and final hidden layer sizes for the iRSM trained and hybridly trained iRSM (iRSM_hybrid) for beta values from 1.1 to 2. Additionally, average RSM score are indicated as lines in the plot for models with hidden layer sizes of 25, 50, 100 and 250 hidden units.

The plots show that a higher beta value results in smaller sized hidden layers. This is the expected behavior since a higher penalty term increases the model's growing threshold. For lower beta values, i.e., lower than 1.2, the size of the hidden layer falls between 75 and 100 hidden units, whereas for higher values of beta, i.e., greater than 1.5, hidden layer size average between 25 an 50 hidden units. Overall, the range of hidden layer sizes of the iRSM seems to be in a reasonable range considering that the best performing RSM has as well 50 hidden units. Hidden layer sizes of iRSM and iRSM_hybrid is very similar which is not surprising considering that the iRSM does not tend to change its hidden layer size much in later stages of the training and since iRSM_hybrid is an iRSM for the first half of the training process its hidden layer size is almost equal to hidden layer sizes of iRSM models.

The top plot of Figure 2, depicting average perplexity scores, shows that the iRSM does not reach the performance of any of the RSM models. The iRSM scores improve with higher values of beta which seems to be correlated with the resulting smaller sized hidden layers. It becomes as well apparent that the iRSM suffers from slightly higher variances (indicated by the shaded areas) than the RSM, which is certainly caused by the non-static hidden layer sizes. The plot shows as well that the hybridly trained iRSM (IRSM_hybrid) performs better than the iRSM and as well better than all RSMs, for the whole range of evaluated beta values.

Table 1 illustrates the influence of training time on the performance of the different models. It can be seen that at epoch 200 RSM models perform better than the iRSM based models. Among the base RSM models, the RSMs with 50 hidden units show the best performance of all throughout the course of the training.

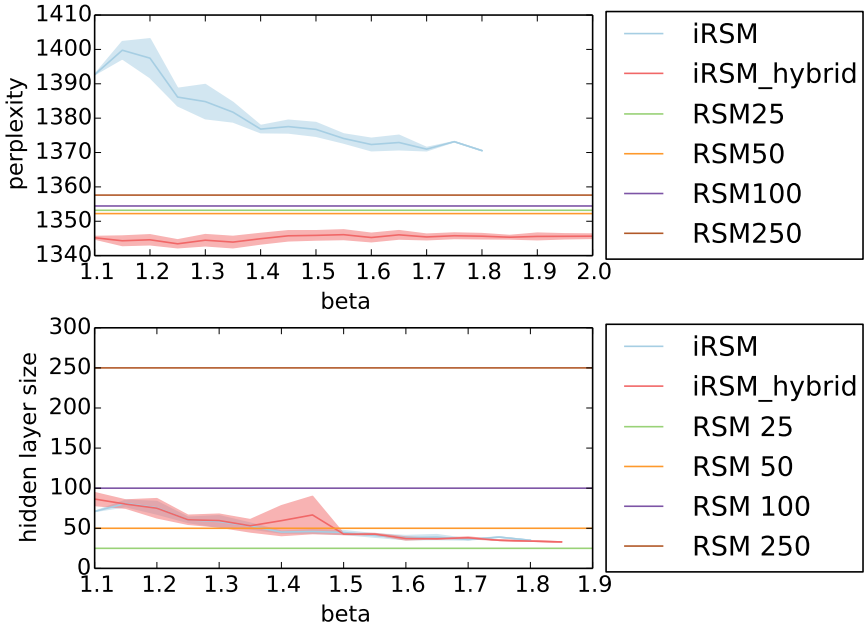


Fig. 2: (top) The plot depicts the average perplexity results of several RSM models with hidden layer sizes of 25, 50, 100 and 250; mean and standard deviation of iRBM models trained with different β settings as well as mean and standard deviation of hybridly trained iRSMs. Mean (line) and standard deviation (shade) based on 10 evaluations per configuration are plotted. (bottom) Mean (line) and standard deviation (shade) of the hidden layer sizes. Results of 10 evaluations per setting. All models have been trained for 800 epochs.

In the interval from epoch 200 to epoch 400 the hybridly trained iRSM has had the largest improvement of all models.

From epoch 200 on all RSM models gradually increase their performance scores as training progresses. Although, the best iRSM models do as gradually increase their performance as the RSM models its performance wrt. the average overall considered beta value decreases. This is causes by the poor performance of the very low valued beta settings which accumulate too many hidden units during the course of the training.

Figure 3 shows the perplexity for 50 randomly selected test documents of RSM, iRSM and iRSM trained in the proposed hybrid manner to give a closer look at how models compare with to each other. The given iRSM and hybrid iRSM models were trained with beta set to 1.5, the RSM model has a hidden layer size of 50 and all models were trained for 800 epochs. The left plot shows that the iRSM perplexity scores are consistently higher than the RSM for all

Table 1: Results of iRSM models for different beta settings compared to RSM models with different hidden layer sizes.

	Avg. Test perplexity per word (in nats)							
	RSM (by hidden layer size)				iRSM (by beta)		iRSM_hybrid (by beta)	
Epochs	25	50	100	250	best	all	best	all
200	1361	**1358**	1360	1363	1375	1390	1375	1390
400	1357	1355	1357	1360	1373	1385	**1346**	1348
600	1355	1353	1355	1359	1373	1388	**1344**	1345
800	1353	1352	1354	1358	1374	1388	**1343**	1344

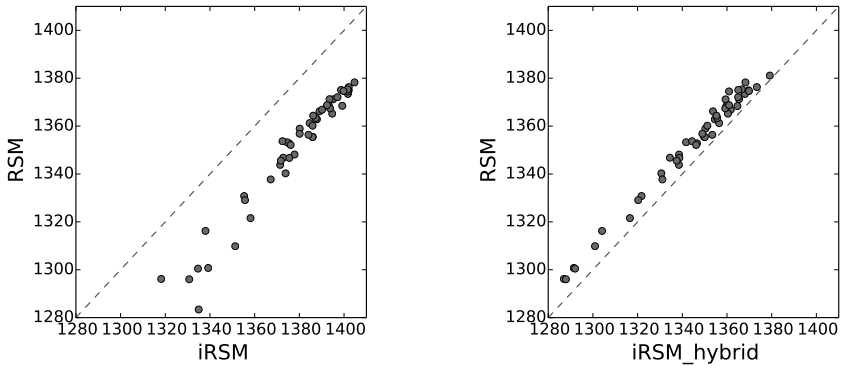
test documents. Similarly, the right plot shows that the hybridly trained iRSM reaches lower perplexity scores than the RSM model.

8 Discussion

The final number of units in the iRSMs adaptive hidden layer seems reasonable when comparing to the hidden layer sizes of the tested RSM models. Figure 2 depicts that for β values smaller than 1.4 the hidden layer size is between 100 and 50 units whereas for bigger values of β it averages between 50 and 25 units. Remarkably, this change in hidden layer size is within the range of the three best performing RSM models under test. These results show that the iRSM is able to adequately adapt its hidden layer size to the documents and reach a reasonable hidden layer size for a broad range of β values.

The results of the conducted experiments summarized in Table 1 and Figure 2 show that the RSM models do reach better perplexity scores early into the training process than the adaptive iRSM models. This is most likely caused by the fact that the iRSMs first have to adapt the size of their hidden layers, from initially 1 unit, on by gradually growing their hidden layers in the first epochs of the training process. Therefore, they suffer from a slow start with respect to representation learning compared to the RSM based models which have all hidden units available to train from the very start. Especially, the hybridly trained iRSM does quickly surpass the performance of the RSM models.

The increase in performance is well depicted in Figure 2. The hybrid iRSM achieves superior scores than both the iRSM and the RSM. A possible explanation for this might be that the RSM models do make steady but small improvements throughout the learning process. The hybridly trained iRSM, on the other hand, does not have this steady monotonous perplexity improvements: it starts rather slow, as already discussed above, but as soon as the transformation to an ordinary RSM takes place the model is able make a big leap wrt. its perplexity



(a) Document-wise perplexity comparison of an RSM with 50 hidden units and an iRSM.

(b) Document wise perplexity comparison of an RSM with 50 hidden units and an hybridly trained iRSM.

Fig. 3: Perplexity score comparison of RSM, iRSM and hybridly trained iRSM on each of the 50 randomly selected test documents. All models were trained for 800 epochs.

scores, see Table 1. The ordinary RSM seems to be able to improve upon the essentially, by the iRSM, pretrained model much better than by starting from a normal initialized RSM model, given that the iRSM learned a reasonable sized hidden layer.

Despite the difference in performance values, Figure 3 as well depicts that all three models seem to, represent each document almost equally well relative to their individual performance realm, which is indicated by the fact that the dots are arranged along an imaginary straight line. One would maybe expect different kind of models to showcase differing representational behavior here, e.g. being able to represent some pattern better than others, and therefore expect a more diffused score pattern. But considering that all the iRSM models inherited especially their representational characteristics from the base RSM, this behavior seems reasonable.

9 Conclusion

In order to adapt automatically the number of topics found in a corpus of text, this paper presents a novel combination of two RBM based methods: the RSM and the iRBM. The resulting iRSM inherits the topic modeling properties of the RSM as well as the iRBM's adaptive hidden layer, which obviates the need to set the size of the hidden layer manually. In addition, we also introduce a hybrid training procedure to effectively increase the performance of the iRSM over the standard training procedure. We conducted empirical experiments to showcase the functioning of the proposed method.

In upcoming work, we are interested in comparing the proposed model with already existing topic models that are as well able to adapt the number of topics, such as the Hierarchical Dirichlet Process Model [15]. For future extensions of this method we are interested in moving from a flat representational structure to structures consisting of several layers of units, that could enable a beneficial interaction between topic features, as already discussed by Salkhutdinov and Hinton [12]. Similarly, Peng et al. developed the infinite deep Boltzmann machine (IDBM) by stacking a fixed number of iRBMs [11].

References

1. Blei, D.M.: Probabilistic topic models. Communications of the ACM **55**(4), 77–84 (2012)
2. Côté, M.A., Larochelle, H.: An infinite restricted Boltzmann machine. Neural computation (2016)
3. DiMaggio, P., Nag, M., Blei, D.: Exploiting affinities between topic modeling and the sociological perspective on culture: Application to newspaper coverage of US government arts funding. Poetics **41**(6), 570–606 (2013)
4. Duchi, J., Hazan, E., Singer, Y.: Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research **12**(Jul), 2121–2159 (2011)
5. Hinton, G.E.: Training Products of Experts by Minimizing Contrastive Divergence. Neural Computation **14**(8), 1771–1800 (2002)
6. Hinton, G.E., Salakhutdinov, R.R.: Replicated softmax: an undirected topic model. In: Advances in neural information processing systems. pp. 1607–1614 (2009)
7. Huhnstock, N.A., Karlsson, A., Riveiro, M., Steinhauer, H.J.: On the behavior of the infinite restricted Boltzmann machine for clustering. In: Proceedings of the 33rd Annual ACM Symposium on Applied Computing. pp. 461–470. ACM (2018)
8. Larochelle, H., Lauly, S.: A neural autoregressive topic model. In: Advances in Neural Information Processing Systems. pp. 2708–2716 (2012)
9. Larochelle, H., Murray, I.: The neural autoregressive distribution estimator. In: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics. pp. 29–37 (2011)
10. Mohr, J.W., Bogdanov, P.: Introduction-Topic models: What they are and why they matter. Poetics **41**(6), 545 – 569 (2013), topic Models and the Cultural Sciences
11. Peng, X., Gao, X., Li, X.: An infinite deep Boltzmann machine. In: Proceedings of the 2Nd International Conference on Compute and Data Analysis. pp. 36–41. ICCDA 2018, ACM, New York, NY, USA (2018)
12. Salakhutdinov, R., Hinton, G.: Semantic hashing. International Journal of Approximate Reasoning **50**(7), 969–978 (2009)
13. Smolensky, P.: Information processing in dynamical systems: Foundations of harmony theory. Parallel Distributed Processing Explorations in the Microstructure of Cognition **1**(1), 194–281 (1986)
14. Srivastava, N., Salakhutdinov, R., Hinton, G.: Modeling documents with a deep boltzmann machine. In: Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence. pp. 616–624. AUAI Press (2013)
15. Teh, Y.W., Jordan, M.I., Beal, M.J., Blei, D.M.: Sharing clusters among related groups: Hierarchical Dirichlet processes. In: Advances in neural information processing systems. pp. 1385–1392 (2005)