ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

Modeling Sparse Data as Input for Weightless Neural Network

Luis Filipe Kopp[1], José Barbosa da Silva Filho[1,2],
Claudio Miceli de Farias[1], and Priscila Machado Vieira Lima[1]

1- Postgraduate Program in Informatics (PPGI) - UFRJ
Av. Athos da Silveira Ramos, 149, Rio de Janeiro, RJ - Brazil

2- Admiral Graça Aranha Instruction Center (CIAGA) - Brazilian Navy
Av. Brasil, 9020, BL I-101, Electricity Lab, Olaria, Rio de Janeiro, RJ - Brazil

Abstract. Dealing with large and sparse input data has been a challenge to machine learning algorithms. In Natural Language Processing (NLP), such challenge is typically faced by bag-of-word solutions wherein the number of useful words is a tiny fraction of the size of the dictionary, leading to sparse input matrices. In this paper we propose aggregating features into groups at random - a simple method for coping with sparse inputs to Weightless Neural Networks (WiSARD) that would reduce the input size. As result, in the considered datasets, we found that bundles of size between 3 to 6 words are typically optimal, and yield an increase of accuracy of up to 4.5%.

1 Introduction

Dealing with large and sparse input data has been a challenge to machine learning algorithms. In Natural Language Processing (NLP), such challenge is typically faced by bag-of-word solutions wherein the number of useful words is a tiny fraction of the size of the dictionary, leading to very sparse input matrices, which may artificially increase computational complexity and decrease model accuracy. For this reason, it is important to identify and cope with input sparsity for model robustness and simplicity.

There are a number of approaches to deal with input sparsity in NLP problems, such as transforming to lower case, removing punctuation and stop-words, stemming, and lemmatization. Feature bundling is also one of those approaches [1, 2] and consists of aggregating multiple features into a single bundled feature. To the best of our knowledge, it has not been combined with the Weightless Neural Network (WNN) framework. WNN have been previously considered for NLP, for the purposes of classifying single phrases or small texts [3, 4]. Currently, many applications involve long texts, such as crime report classification, credit analysis and bias identification on news. WNN had not been efficient for those applications in the past since using traditional methods for reducing the size of the bag-of-words still resulted in very large and sparse input matrices and retinas [5, 6]. The current literature in WNN lacks methods for dealing with large retinas.

In this paper, our goal is to assess the implications of input sparsity for NLP within the WiSARD framework. To that aim, we propose a simple method for

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0. Available from http://www.i6doc.com/en/.

coping with sparse inputs to Weightless Neural Networks (WiSARD). We considered the simplest setup wherein features are aggregated into groups uniformly at random.

We empirically discovered that accuracy of WiSARD increases when bundling words at random. The initial reduction in the complexity of the input builds robustness to the model. Thus, in this study, we propose a new approach to reduce the size of the retina, and we tested the proposed method in a Kaggle challenge database[1]. In particular, we measured how feature bundling impacted classification accuracy. It is worth noting that due to its simplicity, WiSARD is typically used for online solutions wherein training time is a key parameter.

In Section 2 we briefly describe Weightless Neural Networks and the WiSARD implementation. Then, in Section 3 we discuss our data collection, benchmark definition and data analysis. In Section 4 we compare the proposed method against benchmarks and present the results, conclusions, and future directions for future work.

2 WiSARD

Bledsoe and Browning [7] proposed Weightless Neural Network (WNN) in 1959. Standard multilayer feed-forward neural network stores knowledge in the form of network weights whereas WNNs store knowledge in random access memories (RAMs) [8]. WNNs are memory-oriented Artificial Neural Networks for pattern recognition applications. **Wi**lkes, **S**tonhan and **A**leksander **R**ecognition **D**evice (WiSARD) was proposed in 1984 [9], as a Weightless Neural Network model which aims at recognizing patterns represented as binary data.

3 Experimental setup

In this Section, we present how the dataset was collected and treated before applying it to the WiSARD. Then, we discuss the benchmark definition and how the experiment was carried out.

3.1 Data Collection and pre-treatment

We considered to work in this study with funding proposals from the Donorś Choose challenge in Kaggle.[2] In this challenge, some teachers submit proposals for funding and the DonorsChoose.org platform must classify them into “approved" or “disapproved", and if approved, the proposal goes to the platform to gather donations from public worldwide. The challenge is to reduce the amount of proposals that needs human attention, using machine learning algorithms and we faced this challenge using WiSARD Weightless Neural Networks. The proposals are divided into 8 groups (Applied Learning, Literacy & Language, Special Needs, History Civics, Math & Science, Health & Sports, Music & Arts,

[1] Available on https://www.kaggle.com/donorschoose

[2] More information on `https://www.donorschoose.org/`

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0. Available from http://www.i6doc.com/en/.

and Warmth Care & Hunger) and each proposal may be fitted into one or more categories. The categories varied from 10,516 (Special Needs) to 570,150 (Literacy & Language). The categories' division were kept because the words are more similar within than between categories. All our classifiers are run separately per category.

As our goal is to investigate NLP solutions, we used only the free text description column of the project, and information, such as total cost, information related the school, teacher that elaborated the proposal and price of material requested, were ignored for the purpose of this study.

The first step of the treatment process was transforming the free text description to lower case, removing punctuation and stop-words, stemming, and lemmatization, through the use of Natural Language Toolkit (NLTK) library [10, 11]. Then, we removed numerical digits and words of size less than or equal to 3 letters. The bag-of-words model is commonly used in methods of document classification where the occurrence of each word is used as a feature for training a classifier, and is expected to remove insignificant words for classification and occurring equally, both in the approved and unapproved groups.

The resulting words were organized in a bag-of-words matrix. For each proposal we obtained an input vector that was fed to the WNN with an entry equal "0" if the word was not present and "1" otherwise. The output is "1" if the proposal was accepted by DonnorsChoose.org and "0" if not. From over 100,000 words in the dictionary, only 16,000 words were effectively used. Therefore, we obtained a matrix with 16,000 features (columns) and 182,000 samples (rows). Each sample (row) corresponds to a proposal.

The considered dataset poses its inherent challenges. First, after manual inspection we identified for some similar proposal descriptions the approval condition was not consistent. We hypothesize that this is because either (*i*) the proposal with minor corrections was resubmitted, until being approved, (*ii*) an approved proposal is resubmitted by the same author in another round of request for proposals, and at that time it is rejected, or (*iii*) proposals are copiedsince approved ones are public available but it is rejected due to other factors.

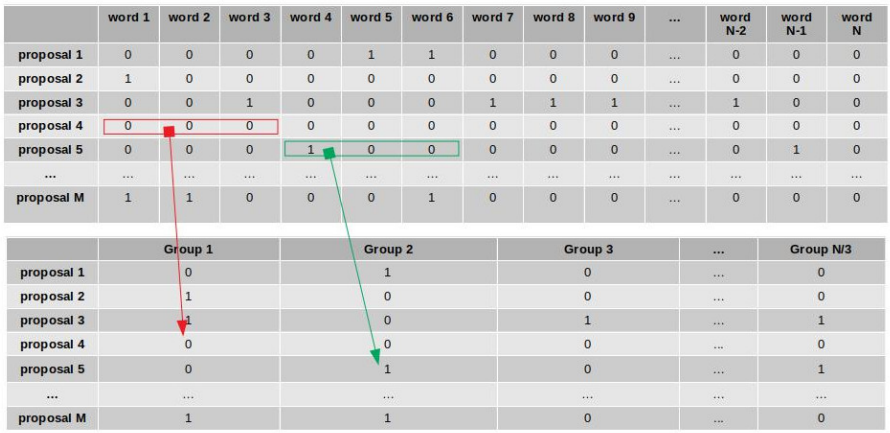
Another relevant problem identified for text categorization in description field is related to the sentiment and emotional appeal of the proposal. This subjective evaluation of text is difficult to be captured by a classification algorithm. Even for humans, deciding by the approval of the proposal based on the description is not precise. Thus, also leading to inaccuracy for the training.

3.2 Experiment description

Our proposal is to group randomly selected features (columns) into a single new feature. The new feature is populated by extracting the maximum value within the elements of the corresponding group. As we considered binary features (corresponding to the presence of a word in a text) the aggregated feature equals one if any of the corresponding words is present in the text, and equals zero otherwise. The number of columns aggregated into a new bundled column varied from 1 (the benchmark) up to 20. Varying the bundling factor, we aim at

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

determining the optimal aggregation level. Figure 1 shows an example of the aggregation method for the input data to the WiSARD.



	word 1	word 2	word 3	word 4	word 5	word 6	word 7	word 8	word 9	...	word N-2	word N-1	word N
proposal 1	0	0	0	0	1	1	0	0	0	...	0	0	0
proposal 2	1	0	0	0	0	0	0	0	0	...	0	0	0
proposal 3	0	0	1	0	0	0	1	1	1	...	1	0	0
proposal 4	0	0	0	0	0	0	0	0	0	...	0	0	0
proposal 5	0	0	0	1	0	0	0	0	0	...	0	1	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
proposal M	1	1	0	0	0	1	0	0	0	...	0	0	0

	Group 1	Group 2	Group 3	...	Group N/3
proposal 1	0	1	0	...	0
proposal 2	1	0	0	...	0
proposal 3	1	0	1	...	1
proposal 4	0	0	0	...	0
proposal 5	0	1	0	...	1
...	...	...	...	...	...
proposal M	1	1	0	...	0

Figure 1: Procedure for aggregating columns. In this example, it is consolidating 3 columns into one.

We randomly selected 2,000 proposals within each category, and split them into training and test sets. Training sets comprised varying from 1/16 to 1/2 of the dataset per category (the remaining entries being used for testing). We repeated the operation 30 times for each of the 8 categories.

Our main metric of interest is the *accuracy gain*. The accuracy gain is the number of correct estimates using aggregated columns minus the number of correct estimates under baseline, then divided by the number of samples.

To further illustrate the rationale behind feature bundling, let p (resp., $1-p$) be the probability that a position in the retina equals 1 (resp., 0). Consider a RAM with two positions, indexed by a single bit chosen uniformly at random from the retina. If $p \approx 0$, position 0 of the RAM will very likely be the single set position. Consider now the bundling of N retina positions. After bundling, positions 0 and 1 will be set with probabilities $(1-p)^N$ and $1-(1-p)^N$, respectively. If $N = \log(0.5)/\log(1-p)$, the two RAM positions will be set with equal probability, increasing the entropy in the state of the RAM which will likely translate into increased information acquired during training. This example allows us to appreciate the potential advantages of feature bundling.

Note that retina feature bundling is analogous to "zooming out" an image and its effect is twofold: (i) the number of pixels (features) reduces and (ii) the essential aspects of the image are captured in a compact fashion.

In this paper we use an implementation of WiSARD with bleaching developed in C and wrapped in a Python library [12].[3] All the experiment was executed in Python 3.6.5, running on a i5 processor with 8Gb of RAM.

[3]Source code available at `https://github.com/IAZero/wisardpkg`

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0. Available from http://www.i6doc.com/en/.

4 Results

We ran the WiSARD, being the training set 1/2, 1/4, 1/8, and 1/16 of the 2,000 proposals randomly selected for each of the 8 categories. We compare it against features aggregated uniformly at random in groups of 3, 6, 10, and 20 features. Table 1 reports the average percentage of accuracy gain compared against the benchmark, for training set comprised of 1,000 proposals. The overall baseline accuracy was 74.2% and the accuracy gain on average was +3.8%. The aggregation of 3 and 6 columns resulted in higher levels of accuracy gain up to +4.5%. Category "Warmth Care" posed significant classification challenges. This may be because its baseline accuracy was already over 90% and our approach had only marginal improvement.

	Number of Aggregated Columns			
Proposal Categories	**3**	**6**	**10**	**20**
Applied Learning	+5.4%	+6.4%	+4.8%	+2.6%
Literacy & Language	+5.0%	+4.8%	+4.2%	+4.0%
Special Needs	+4.9%	+5.3%	+5.6%	+2.2%
History & Civics	+4.8%	+5.0%	+3.0%	+2.4%
Math & Science	+7.0%	+6.0%	+4.8%	+3.0%
Health & Sports	+4.3%	+3.9%	+3.7%	+1.6%
Music & Arts	+4.2%	+3.4%	+3.2%	+2.3%
Warmth Care & Hunger	+0.7%	+0.7%	+0.6%	+0.2%

Table 1: Improved accuracy of aggregated columns compared to benchmark.

Next, we formally verify the statistical significance of our results. We conducted a pairwise t-test. Figure 2 shows that across all simulations, on 89.75% of them feature bundling over-performed the benchmark. We set as our null hypothesis that the accuracy gain with bundled features is zero. Then, we were able to reject the null hypothesis ($t(1143) = -27.95$, p-value < 0.001), which implies that feature bundling produces positive accuracy gain.

5 Conclusion

In this paper we presented preliminary results on the use of feature bundling for NLP under the Weightless Neural Network framework. Our results indicate that feature bundling may increase classification accuracy and reduce input complexity. Feature bundling for NLP should account for word proximity as blurring accounts for object contours in image processing.

This work opens up several directions for future investigation. First, we must identify better ways of grouping features, e.g., taking into account semantic value of the word. Second, we suggest to evaluate feature bundling together with other machine learning tools, and with other sparsity levels.

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

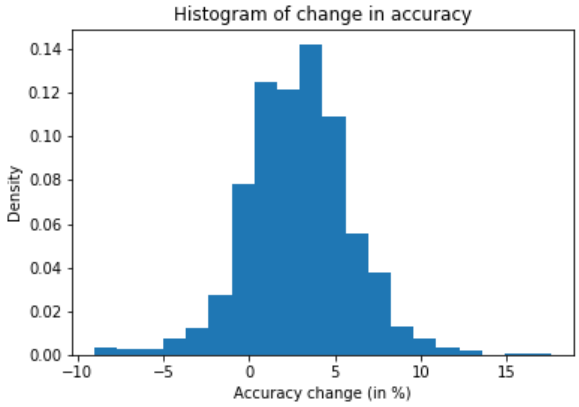


Figure 2: Histogram of the variation on accuracy due to the method proposed.

References

[1] Z. Wu, Q. Ke, M. Isard, and J. Sun. Bundling features for large scale partial-duplicate web image search. In *Computer Vision & Pattern Recognition*, pages 25–32. IEEE, 2009.

[2] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *NIPS*, 2017.

[3] C. L. R. Motta R. A. Pinho, W. A. T. Brito and P. V. Lima. Automatic crime report classification through a weightless neural network. *ESANN*, pages 165–170, 2017.

[4] M. de Gregorio R. D. Cavalcanti, P. M. V. Lima and D. S. Menasche. Evaluating weightless neural networks for bias identification on news. In *ICNSC*, pages 257–262, May 2017.

[5] G. Mamakis, A.G. Malamos, and J.A. Ware. An alternative approach for statistical single-label document classification of newspaper articles. *Information Science*, 37(3):293, 2011.

[6] Nadia Al-Bakri and Soukaena Hashim. Reducing data sparsity in recommender systems. *Al-Nahrain Journal of Science*, 21(2):138–147, Sep. 2018.

[7] W. W. Bledsoe and I. Browning. Pattern recognition and reading by machine. In *Computer Conference*, volume 203, pages 225–232. ACM Press, 1959.

[8] A.F. de Souza, F.D. Freitas, and A.G.C. de Almeida. High performance prediction of stock returns with VG-RAM weightless neural networks. In *WHPCF*, pages 1–8, 2010.

[9] I Aleksander, W.V. Thomas, and P.A. Bowden. WISARD: a radical step forward in image recognition. *Sensor Review*, 4(3):120–124, mar 1984.

[10] Edward Loper and Steven Bird. Nltk: The natural language toolkit. In *Effective Tools for Teaching Natural Language Processing & Computational Linguistics*, 2002.

[11] David A. Hull. Stemming algorithms: A case study for detailed evaluation. *J. Am. Soc. Inf. Sci.*, 47(1):70–84, January 1996.

[12] F. França D.S. Carvalho, H.C.C. Carneiro and P. Lima. B-bleaching: Agile overtraining avoidance in the wisard weightless neural classifier. *ESANN*, pages 515–520, 2013.