Available online at www.sciencedirect.com

ScienceDirect

Cognitive Systems Research 45 (2017) 59–69

Cognitive Systems RESEARCH

www.elsevier.com/locate/cogsys

Deep learning and punctuated equilibrium theory

Simon Hegelich

Bavarian School of Public Policy, Technical University of Munich, Richard-Wagner-Str. 1, D-80333 Munich, Germany

Received 21 June 2016; received in revised form 11 December 2016; accepted 28 February 2017
Available online 11 April 2017

Abstract

Deep learning is associated with the latest success stories in AI. In particular, deep neural networks are applied in increasingly different fields to model complex processes. Interestingly, the underlying algorithm of *backpropagation* was originally designed for political science models. The theoretical foundations of this approach are very similar to the concept of *Punctuated Equilibrium Theory* (PET). The article discusses the concept of *deep learning* and shows parallels to PET. A showcase model demonstrates how deep learning can be used to provide a missing link in the study of the policy process: the connection between attention in the political system (as inputs) and budget shifts (as outputs).

Keywords: Deep learning; Neural networks; Punctuated equilibrium; Policy process; Backpropagation

1. Introduction

Deep learning is associated with the latest success stories in AI. From autonomous cars to AI beating a Go-master: deep learning is the method of choice to construct machine learning models that are useful in many complex situations. Taking this success-story into account, it seems obvious that political science could profit from this method as well. A whole branch of political science approaches sees the *policy process* as a kind of cognitive system that transforms political inputs from society into outputs. *Punctuated Equilibrium Theory* (PET) is a very successful concept in political science that is grounded in the theoretical works of Herbert Simon on *bounded rationality* (Simon, 1955). What separates this approach from rational choice theories is - amongst others - the understanding of organizations (Jones, 2003). While theories that rely solely on market mechanisms can see organizations only as individuals (maximizing their utility function) or as markets, where individuals meet. In bounded rationality organizations are seen as cooperations of individuals identifying with the organization. In many cases, this makes organizations much more effective than markets, especially because parallel processes can be organized with less information costs. But this makes organizations quite complex, as well.

> If complex systems must operate in a constantly changing environment [. . .] they must modify their structures at a corresponding pace. The need for close coordination, even in the presence of strong identification with the organization's goals, places a very heavy burden on a system's capacity to evolve toward greater effectiveness under changed conditions. For although identification reduces the need to police self-interest and to ensure its compatibility with organizational objectives, it also causes excessive influence of existing organizational practices and identifications upon decisions that should be adapting to a changing world (Simon, 2000, p. 753).

E-mail address: simon.hegelich@hfp.tum.de

http://dx.doi.org/10.1016/j.cogsys.2017.02.006
1389-0417/

Deep neural networks are very suitable for dealing with this complexity. Taken into consideration the strong interest of Herbert Simon in machine learning, it seems surprising that the connection between PET and deep learning has not yet been tapped for the benefit of political science, at least to the best of my knowledge. This is even more astounding against the background that the underlying algorithm of *backpropagation* was originally designed for political science models. Unfortunately, the formerly mutual connections between computer science and political science seem today to have eroded. Cognitive system research seems to be an ideal place, to bridge this gap between the two disciplines by highlighting the parallels of deep learning and PET. Hopefully, this attempt will work as a humble contribution to re-establish the interdisciplinary field of *political data science*.

The article proceeds as follows: First, the concept of deep learning is introduced with a focus on neurons as the building blocks of neural nets. Second, the idea to understand the policy process as information processing is recapitulated and linked to the problems of complexity and noise. On this basis, third, a showcase of the implementation of deep learning in PET is presented. It is demonstrated, that deep learning is capable of linking attention signals in the political system to policy outcomes in the form of budget changes. Fourth, the theoretical relevance of this demonstration is discussed. Finally, the article provides an outlook explaining how more advanced deep learning models could push the development of PET even further.

2. What is deep learning?

Deep learning as a machine learning approach that is based on *neural networks*. A very good description of neural networks is given by Pat Langley and Herbert Simon:

> One major paradigm, associated with the area of neural networks, represents knowledge as a multilayer network of units that spreads activation from input nodes through internal units to output nodes. Weights on the links determine how much activation is passed on. The activations of output nodes can be translated into numeric predictions or discrete decisions about the class of the input. [...] One common learning algorithm, among the many that have been explored, carries out gradient descent search through the space of weights, modifying them in an attempt to minimize the errors that the network makes on training data (Langley & Simon, 1995).

There are different opinions which conditions make a neural network actually "deep". A very basic idea of deep neural networks is the combination of *multiple hidden layers* of non-linear transformations. In the show-case model used in this article for demonstration this basic definition of deep learning is applied. "Real" deep learning goes far beyond the simple addition of layers in neural networks but alters the underlying algorithms to create feedback loops and reinforcement learning. These aspects will be discussed in Section 6.

In an analogy to biological processes in the brain the building blocks of neural networks are called "neurons". A neuron collects different inputs and transfers them into a non-linear output. From a mathematical point of view, a neuron combines two functions: a summation function $f(s)$ and an activation function $f(a)$. "Given a sample of input attributes $x_1, \ldots, x_n$ a weight w_{ij} is associated with each connection into the neuron" (Lewis, 2016, p. 16). All inputs are summed up according to:

$$f(s) = \sum_{i=1}^{n} w_{ij} x_j + b_j \tag{1}$$

The parameter b_j is the bias and can be interpreted like an intercept in regression. It allows to the activation function to be shifted upwards or downwards.

The activation function takes the result of the summation as an input and transfers it in a non-linear way, usually to values between 0 to 1 or -1 to $+1$. There are many different functions that can be used as activation function but the s-shaped sigmoid function is very common (Friedman, Hastie, & Tibshirani, 2001, p. 393):

$$f(a) = sig(t) = \frac{1}{1 + e^{-t}} \tag{2}$$

The reason for the popularity of the sigmoid function is that it can be differentiated very easily, which is important for the optimization process in neural networks.

A neural network is a combination of single neurons. In a deep neural network the output from a layer of neurons functions as input for the next layer (see Fig. 1).

It is import to note that the strength of the connections of the neurons is determined by the weights assigned in the summation function of the following neuron. An output from a neuron (or the input layer) may have weight 0 in the summation function of one following neuron - i.e. it does not count for activation at all. But for another neuron the same output may have a higher weight that makes it very important for activation. Two things should become clear at this point:

1. A deep neural net is able to represent a very complex non-linear prediction space.
2. The final result of the model is determined by the assigned weights of the neurons.

This leads to the question: How are the weights assigned?

The basic algorithm to calculate the weights is called *backpropagation*. "The first practical application of backpropagation was for estimating a dynamic model to predict nationalism and social communication in 1974" (Werbos, 1994, p. 270) by Paul J. Werbos in his dissertation. The connection between political science and deep learning

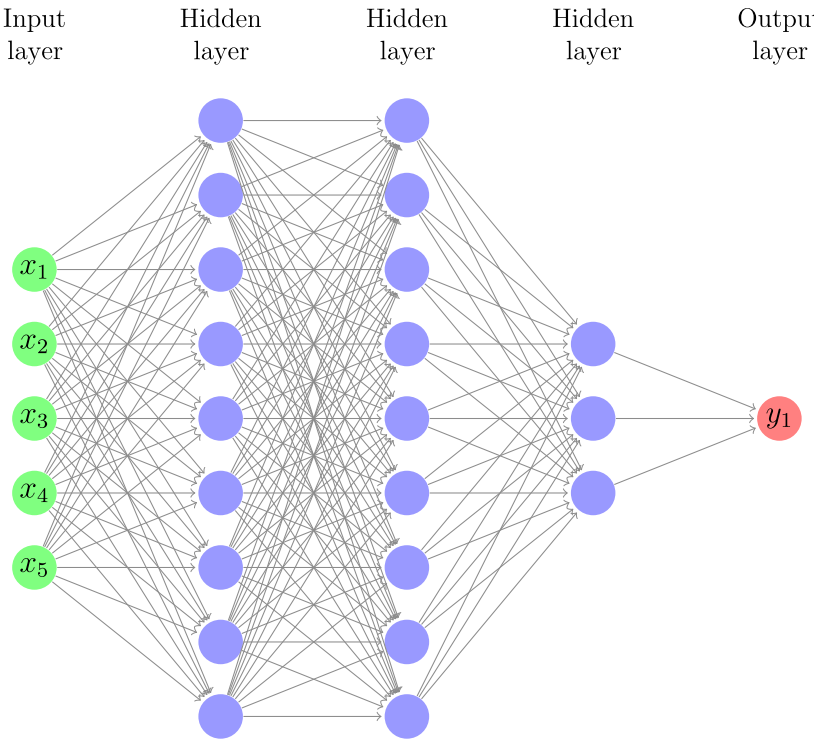

Fig. 1. Deep neural net example.

therefore was very strong in the beginning. But unfortunately, Werbos' work had a much stronger effect on computer science than on political science. The basic idea of the algorithm is quite simple: To begin, there must be some data with inputs and outputs to train the model. First, each neural connection gets a random weight so that the model produces outputs and a square error E can be calculated from the difference of these outputs to the observed values. The innovative idea now is that with these given values we can calculate the derivatives of the activation functions of the neurons (starting with the last and then going *backwards*). Therefore, we know, if the weights of a neuron should be increased or lowered to minimize E (Werbos, 1994, pp. 272–277). This procedure is repeated until an optimum is reached (or in practice, until a stopping rule like "stop after 1500 attempts" is activated). Friedmann et al. provide a mathematical notation and explanation of the backpropagation algorithm (Friedman et al., 2001, p. 396).

The described procedure is a non-parametric method and to this extent similar to other machine learning algorithms like *random forest* or *support vector machine* (see Hegelich, 2016, p. 99). This means, the model is estimated directly from the data instead of estimating parameters for a probability density function (PDF). At first glance this difference may seem trivial, but it is not: In parametric methods it is assumed that there is a data-generating process that is correctly modeled by the underlying function. As long as we stick to this model, we believe that the predicted values are the *true* values while the observed values are random deviations from this truth, caused by *noise*. The plausibility of the model is tested against its accuracy in prediction (normally measured as R^2 or any other derivate of squared or absolute errors). Second, it is tested if the deviance of the observed values from the predicted values is likely to be the product of random noise (significance tests and confidence intervals).

In the following Section 1 will discuss why this *epistemological* point of view - the model (as long as it is accepted) represents the *true* data-generating process and the observed values are *random* deviations from this truth - is very problematic when we are dealing with policy processes. At this point it is important to stress that deep learning is different: The *observed* data-generating process is taken as true. The deep neural network is a mathematical representation of a process that produces results that are as similar to the observed results as possible. It does not matter, if the network with its neurons is a good representation of the real process' structure as long as the results are similar. In fact, it is common practice to build networks that are *over-complex* (that is why the layers are called *hidden*) and then to trust in the ability of backpropagation to shut down unnecessary neurons by assigning zero weights. "Generally speaking it is better to have too many hidden units than too few. With too few hidden units, the model might not have enough flexibility to capture the nonlinearities in the data; with too many hidden units, the extra

weights can be shrunk toward zero if appropriate regularization is used" (Friedman et al., 2001, p. 399).

It is important to note that this *over-complex* structure means that neural networks are not meant to be a simulation of a real world process: AlphaGo, the AI of Google DeepMind that has beaten a Go-master, uses deep neural nets. But nobody would argue that it plays the game *like* a human. Instead it uses a very complex structure to produce *similar* results. To take the observed data as the truth also means that there is no "excuse" for deviations of predicted values and observed values. The whole goal of the backpropagation algorithm is to minimize the prediction error and if at one point the prediction is poor, deep neural nets will try to find a better representation of the data-generating process, even if this changes the whole structure of the network.

3. Policy process as information processing

Jones and Baumgartner (2005) have described the policy process as information processing (see Fig. 2).

Different sources of information are received via indicators by political institutions and agents. Therefore, the policy process is shaped by the capacity of these actors to process this information. But due to bounds of rationality the information processing capacities of individuals and organizations differ fundamentally: while organizations are capable of parallel processing, individuals have to rely on serial processing. This means, two different kinds of processes are going on: while political institutions produce steady outputs based on routines, the decision-makers in a hierarchical structure can focus their attention only on single issues. The results are the repeatedly observed characteristics of PET: incrementalism *and* drastic policy shifts, overreaction *and* underreaction, policy bubbles *and* "negative" policy bubbles, etc. The outputs of the policy process follow a "heavy tailed" distribution, i.e. while the majority of outputs is moderate at any given time, we will find in longer time-series punctuations of this equilibrium; or in a more mathematical formulation: The probability of extreme values is much higher than we would expect from a normal distribution (see Jones, Sulkin, & Larsen, 2003, p. 164).

The resulting distribution of policy outputs can be demonstrated in annual percentage budget changes in the USA (see Fig. 3) and probably any other country. It has been declared as "general empirical law" by Jones et al. (2009) and has been tested in many cases (Breunig & Koski, 2012; Flink, 2015; Jones & Baumgartner, 2012; Jones, Zalányi, & Érdi, 2014). Simulations have shown that this heavy tailed distribution necessarily results from the described information processing structure (Thomas III, 2016). In principle, it should therefore be possible to predict policy outputs by using a heavy tailed PDF in a generalized regression model. Hegelich, Fraune, and Knollmann (2015) go this way and they provide proof of concept in the case of nuclear energy budgets in the USA. Besides using a generalized regression model (general linear model (GLM)) with an extreme value function, they use a very complex set of input variables that is heavily filtered with data-mining techniques. This approach works, but only to a certain extent: The model outperforms models with the same inputs and without heavy tailed PDF as well as models with simpler inputs. But the prediction error E of the model is still very high and its strengths seem to lie in identifying different corridors of uncertainty rather than in delivering sound predictions (Hegelich et al., 2015, p. 250). Nevertheless, the model shows that there is a connection between indicators of attention and actual policy outcomes and that the probability of system shifts can be modeled over time and this goes beyond the stochastic process models of PET (Breunig & Jones, 2011; Breunig & Koski, 2012). But at the same time, the skepticism of Jones towards predictions of budget shifts seems to remain valid:

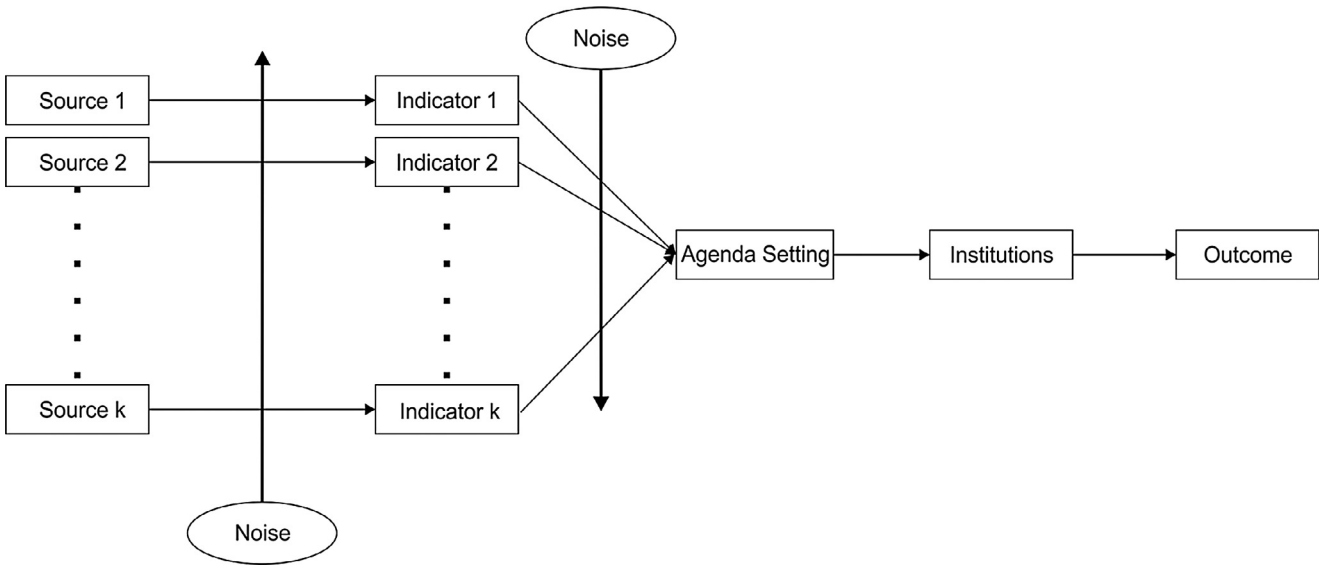

Fig. 2. Information processing. *Source:* Jones and Baumgartner (2005, p. 165)

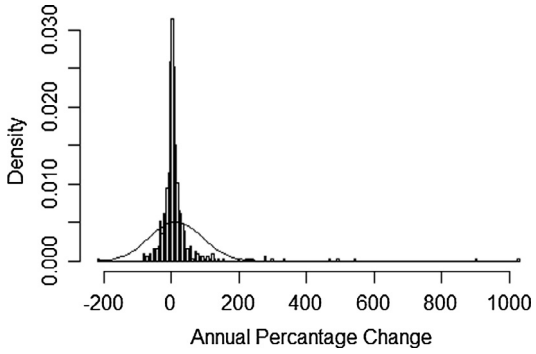

Fig. 3. Histogram of percentage budget shifts. *Source:* Hegelich (2016, p. 102)

> Institutional costs in politics may approximate the manner in which friction operates in physical models. When friction is introduced into idealized physical models, nonlinear systems result [...]. Such open systems result in an output pattern that is episodic and punctuated, with extraordinary difficulty in making point predictions (Jones et al., 2003, p. 154).

From the discussion of deep learning and non-parametric approaches and from the theoretical foundations of PET, we can derive a strong argument that explains success and failure of the GLM approach. The problem is the noise. As can be seen in Fig. 2, the outputs of the system are not only shaped by the described duality of serial and parallel processing, but also by the introduction of noise at different levels. The GLM approach might consider the heavy tailed PDF of the signal, but the deviation of the observed values from the seemingly *true* model is taken to be random, i.e. following a normal distribution. The theory behind this noise-model is the central limit theorem (CLT). The CLT states that the mean of a large number of iterates of *independent* random variables will be approximately normally distributed, regardless of the underlying distribution, in case they have a well-defined expected value and *well-defined variance*. As long as we think of noise in the policy process as the sum of different emitters of disturbance that are *independent* from each other and are producing random but stable (well defined variance) outputs, normal distribution is the safest guess. But Werbos (1994) provides two strong arguments, why these assumptions do not fit to the policy process and to social science in general.

The first argument is questioning the "well-defined variance" statement. If we imagine that there is one emitter of disturbance that leads to very strong deviations but is only very seldom active or even not on a regular basis but only once and then disappears for ever, the noise will not follow the expected normal distribution.

> There may be many processes that normally plod along in a predictable sort of way, governed by a noise process $b(t)$ that fits a normal distribution and that never gets to be very large; every once in a while, however, the process may be hit by a fluke, which leads to changes much larger than one would have expected in the normal course of events. Suppose that p_1 is the probability, at any time, of getting a fluke. [...] most of the time - $(1 - p_1)$ to be precise - b will fit the same bell-shaped curve as before; *however*, when a ``fluke" occurs, b will fit a much broader bell-shaped curve, leading to much larger values for b (Werbos, 1994, pp. 54–55).

From a PET perspective this should sound very familiar. In fact, if we accept political signals to be heavy tailed distributed, it is very unlikely that political noise follows a normal distribution.

The second argument is "measurement noise" (Werbos, 1994, p. 55). It is very unlikely that data in political science is an unbiased representation of the real world. PET accounts for this explicitly by introducing noise between the information sources and the received indicators (see Fig. 2).[1] The problem with measurement noise is that it is excavating the core assumptions of classical statistics: first, it transfers the model - that is taken to represent the *true* data-generating process while the observed values are thought to be random deviates - from an *explicit* model to an *implicit* one.

In an explicit model the current state of a system $U(t)$ allows to calculate the state at a later time $U(t + \Delta t)$ with an error $c(t)$, which - for the sake of the argument - might be normally distributed:

$$U(t + \Delta t) = F(U) + c(t) \tag{3}$$

If we consider process noise *and* measurement noise, we cannot calculate $U(t + \Delta t)$ from $U(t)$, any more. Let's assume we do not measure the *real* state of the system U but only U' where d is the measurement noise.

$$U'(t + \Delta t) = U(t + \Delta t) + d(t + \Delta t) \tag{4}$$

> Given that we do not know the true value of $U(t)$ at any time t, this model is not an ``explicit" model; it does not tell us directly how to estimate $U'(t + 1)$ [$U'(t + \Delta t)$ in the here used notation, N.N.] from earlier available data (Werbos, 1994, p. 55).

The results of the model cannot be taken as true, if the inputs do not represent true values either. Even if the causal relation the model presents was actually true, the statistical model would come up with wrong results. This does not mean that the model will not "work" to predict the observed values. But it means that the model cannot be

[1] To be precise: the noise model illustrated in Fig. 2 introduces "measurement noise" within the policy process: political agents do not receive unbiased information of the world but only *indicators*. Scientists have to try to estimate these indicators and will thereby add another layer of measurement noise.

taken as a representation of the *true* data-generating process. Therefore, it has no higher plausibility then any other model and the *only* justification available is the comparison of predicted and observed data.[2]

Taking these two arguments about the nature of noise in social science seriously, we have three options on how to proceed: we can try to build advanced models with explicit noise models included and with mechanisms to detect measurement noise over time (Box, 1979; Harvey & Todd, 1983). Or, we stop pretending that our models are valid representations of the true data-generating process. Then, we either stop making predictions and concentrate on the stochastic process that reveals its characteristics over time but stays unpredictable (Breunig & Jones, 2011). Or, we stop worrying about the process and concentrate on predictions, assuming that the model with the lowest error E is the best representation of what is going on in our data. Apparently, all these approaches would converge with growing success: very good complex models would steadily reduce the bias from noise. Stochastic process models would reveal more and more hidden characteristics of the real-world process. And deep learning models will generate an artificial process similar *in its outputs but not necessarily in its structure* to the real-world process, because otherwise, predictions would not be stable.

4. A showcase model: deep learning and the policy process

To demonstrate the potentials of deep learning, I will build a model that tries to predict punctuations in budget shifts from the attention of political actors. Data is taken from the *Comparative Agendas Project* (CAP) for the case of the US.[3] Budget and attention data is coded in different systems in CAP, but some of the major functions are quite comparable (e.g. budget function "Natural Resources and Environment" and CAP function "Environment"). Table A.3 shows the corresponding and non-corresponding codes. Fig. 3 shows the histogram of the eleven budget functions (as annual percentage change) for which there are corresponding CAP functions. There are several definitions in the literature on which observations in a distribution qualify as a punctuation and which do not. Baumgartner and Epp (2013) recently suggested counting the top and bottom 5 percent of the observed changes as punctuations. In an earlier study Jones, Baumgartner, and True (1998) qualified annual increases greater than 20 percent and decreases greater than 15 percent as punctuations. Epp and Baumgartner recently proposed a new definition for puncutations (Epp & Baumgartner, 2016, p. 4). Although their *squared folded percentile ranking index* is an improvement compared with simple cutoffs like +20 and −15 per cent, it has some problematic characteristics: first, it finds punctuations per definition. If this index was used for distributions without heavy tails, it would identify punctuations, as well. Second, Epp and Baumgartner construct this index over the combined distributions of all budgets. But we know that different functions follow different distributions - although maybe all of them are heavy tailed (Jones et al., 2009). To evaluate if a budget change can count as punctuation, therefore the *inter quantile range* (IQR) is calculated for each budget function as proposed by Hegelich et al. (2015, p. 240) by the following formular:

$$IQR(x) = quantile(x, 3/4) - quantile(x, 1/4) \tag{5}$$

Any budget change is counted as punctuation if it satisfies the following condition:

$$quantile(x, 3/4) + IQR(x) * 1.5 < x \vee x < quantile(x, 1/4) - IQR(x) * 1.5 \tag{6}$$

This formular transfers the dependent variable (Punc) into the boolean values *TRUE* or *FALSE*. This use of the *IQR* is exactly what is commonly used in parametric statistics to describe "outliers". The idea behind this is that the deep learning model shall predict exactly these points classical methods would classify as "obscure".

The independent variables are constructed from the datasets "Public Laws" (lawsS), "Congressional Hearings" (congS), "Executive Orders" (eoS), "State of the Union Speeches" (souS), and "Gallups Most Important Problem" (gallupS). For each CAP topic that corresponds to a budget function an annual count is created (e.g. if Congress had 10 hearings in 1970 on "defense" the value would be 10). For "Gallups Most Important Problem" the percentage of mentions of the topic in the actual year is taken as value. Data is scaled before combining the different topics. The final dataset covers a timespan from 1950 to 2013 and consists of 580 entries in six columns.

To evaluate the predictive power of different models, the dataset is divided in a trainingset of 386 randomly choosen observations and a remaining set of 194 observations for testing.

A deep neural net with an input layer of five neurons (the independent variables) and three hidden layers of nine, nine and three neurons is fitted on the trainingset with the classical *backpropagation* algorithm (see Fig. 1). In accordance with the "measurement noise" discussion above, different numbers of hidden layers and neurons have been tested to find a combination that leads to the best predictions. It should be noted therefore, that there is nothing special about the specific structure of the network and any variation that would come up with better results was to be preferred. There is no connection between this special structure and the structure of the policy process in the real

[2] In addition, measurement noise cannot be modeled as *independent* emitters because it might effect *every* measurement.

[3] The data used here were originally collected by Frank R. Baumgartner and Bryan D. Jones, with the support of National Science Foundation Grant Nos. SBR 9320922 and 0111611, and are distributed through the Department of Government at the University of Texas at Austin. Neither NSF nor the original collectors of the data bear any responsibility for the analysis reported here.

world. To find an adequate stopping criterion, the squared errors for every epoch of backpropagation are plotted (Fig. 4). After 5000 iterations, the model seems to produce stable results. To be on the safe side, the maximum iteration parameter is set to 6000.

We can now check the performance of the model by evaluating the confusion matrix (Table 1). The results are very promising: the model has rightly labeled all but one data-point of the trainingset. Unfortunately, there is a typical phenomenon in deep learning, called *over-fitting*. The algorithms are sometimes very accurate on the dataset the model is fitted to but perform poorly on new data. The state of the art procedure to deal with this situation is cross-validation. The idea is to build the model on one dataset and test it on a different one. "Ideally, there would be two random samples from the same population. One would be a training data set, and one would be a testing data set. [...] Often, there is only a single data set. An alternative strategy is to split the data up into several randomly chosen, nonoverlapping parts" (Berk, 2006, p. 277). For cross-validation the dataset is split randomly in a training set containing e.g. two thirds of the data and a test set with the remaining one third. The final model is fitted on the training data only and the predictions for the test data are evaluated. This validation set approach in principle should prevent over-fitting. An advantage of this method is that it is easy to apply, but there are two potential drawbacks that should be kept in mind:

1. The validation-set approach can lead to quite different results, depending on the actual division of training and test set. In practice, splitting the data should always be made with a "frozen" random number generator so that others are able to reproduce the results.
2. "Since statistical methods tend to perform worse when trained on fewer observations, this suggests that the validation set error rate may tend to overestimate the test error rate for the model fit on the entire data set" (James et al. 2013, 178). The splitting of the data in a training set and a test set therefore leads to a lower level of accuracy. The two thirds approach is often seen as best-practice, because it takes many observations for training - which leads to high accuracy - but leaves sufficient observations for testing. But in practice, any other proportion of test and training data is possible (e.g. two sets of equal size).

Table 1
Confusion matrix trainingset.

		FALSE	TRUE	Total
			Predicted	
	FALSE	347	0	347
Observed	TRUE	1	38	39
	Total	348	38	386

As described above, data for testing was reserved for testing, so we should take a look at the confusion matrix, where the model is predicting new data.

As can be seen from Table 2 the model is misclassifying some datapoints on new data. But the accuracy is still quite high with 85 per cent. Three of the 14 punctuations in the dataset were predicted correctly, but 18 times the model predicted a punctuation falsely. To work as a reliable classificator for punctuations, the model would have to be improved. But nevertheless, this test-case shows that there is a signal in the data, that can be estimated. An alternative model from parametric statistics would be a generalized linear model based on a binomial distribution. This model was fitted on exactly the same data. But neither in the test-set nor in the trainingset was the GLM able to identify any punctuations in the budget data, because it is averaging predictions to the mean.

Both models not only deliver the pure predicted values, they also include an estimation of how sure the model is about each prediction. These probabilities can be used to visualize the performance of different classification models by plotting them in an ROC curve. "The name "ROC" is historic, and comes from communications theory. It is an acronym for receiver operating characteristics" (James, Witten, Hastie, & Tibshirani, 2013, p. 147). For an ROC curve the predictions are ordered by their probabilities. In some cases the model is very sure that there was no punctuation (e.g. probability 0.1), in other cases TRUE is

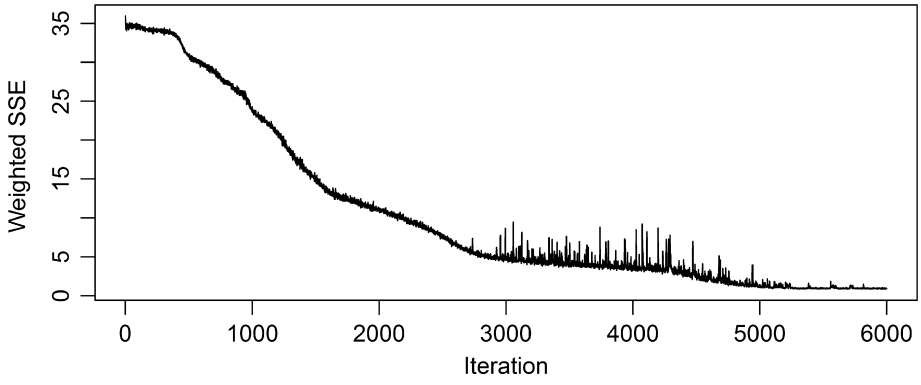

Fig. 4. Backpropagation error.

Table 2
Confusion matrix testset.

		FALSE	TRUE	Total
			Predicted	
	FALSE	162	18	180
Observed	TRUE	11	3	14
	Total	173	21	194

a more likely prediction (e.g. probability 0.8). For every probability value the true positives (i.e. the cases which have been rightly labeled, also called sensitivity) and the false positives (i.e. the cases which have been wrongly labeled also called specificity) are counted. Plotting these values against each other results in an ROC curve (see Fig. 5). A perfect model that predicts every observation correctly would be represented by a ROC curve that hugged the top left corner of the plot. The bigger the area under the ROC curve (AUC), the better the model is. A purely random classifier has an AUC of 0.5 and is represented by a straight diagonal in the plot. Because true positive rate and false positive rate are independent from the type of classification model, we can use ROC curves to compare the performance of any classifier.

Fig. 5 shows that the deep learning approach clearly outperforms the GLM. The latter is not able to identify any signal and performs like a pure random classifyer (or even worse).

5. Linking theories of budgetary politics to the politics of attention

In a nutshell, PET sees the policy process as information processing. Bounded rationality leads to disruptive patterns of attention. The institutional layout of the system adds friction to the information process. The outputs of the system - especially budgets - can be described as *disrupted exponential incrementalism* (Jones et al., 2014, p. 4). There is a sound theory behind this process model: bounded rationality (Jones, 2003; March & Simon, 1958). But empirical research remains somewhat fragmented. There is a lot of work on attention, the different institutional settings in different political systems and the stochastic process of budget functions. Some empirical studies try to connect two of these links in the information processing

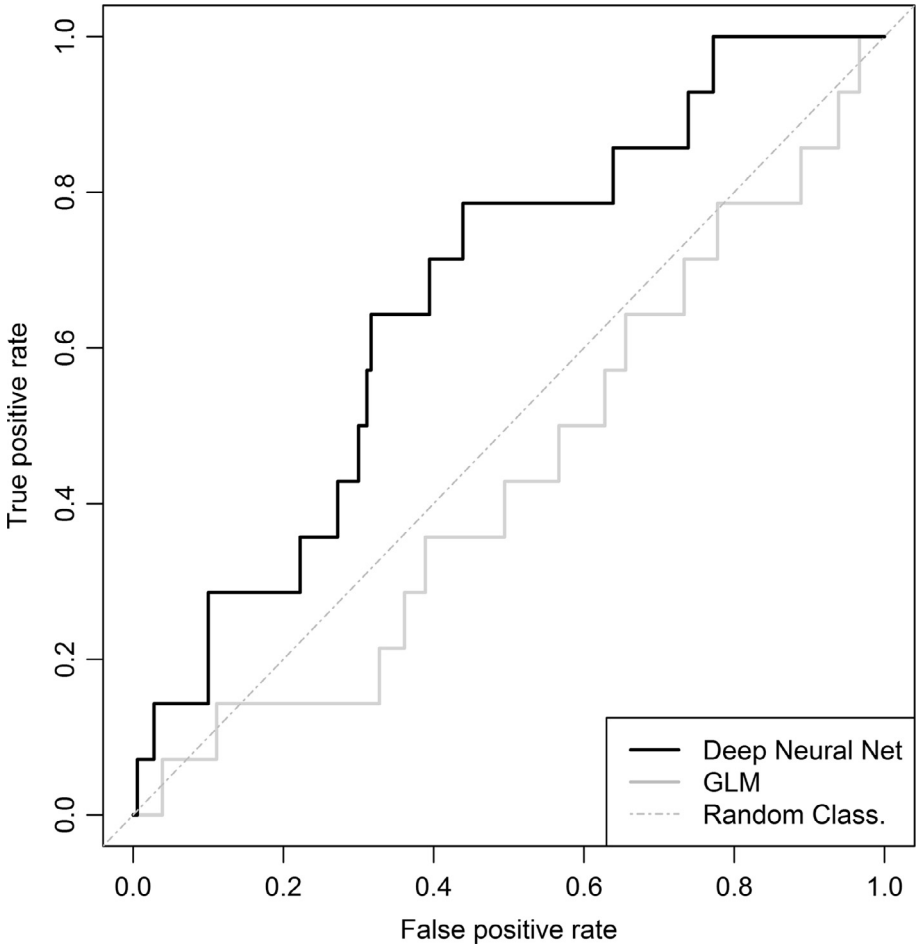

Fig. 5. ROC curve.

chain: attention and institutions, institutions and budgets. But studies that link the attention inputs to the budget outputs are very rare.[4] But as long as only fragments of the framework are tested individually, the broader concept remains questionable. Without providing empirical evidence for the whole chain of information processing, the underlying causal explanations of PET are unstable. For example, John and Bevan (2012) argue for "minor punctuations" that are not connected to attention. Howlett (2009) criticizes the lack of causality because he is focusing only on the work on the stochastic process. There are at least two reasons why it is very difficult to provide empirical evidence for the hypothesis that the policy process is a system of information processing: one is noise, the other is complexity.

Recent studies in PET provide new insights as to the importance of the above described forms of "process noise" and "measurement noise". Jones et al. (2014) find that even when we consider exponential shifts in budgets, long time series reveal "critical junctures". There are disruptive breaks in policy process like wars and major crises. As Werbos (1994) explains, this affects the distribution of noise so that it is nearly impossible to build an explicit model. Measurement noise is addressed by John and Bevan (2012) in their concept of "procedural punctuations". If a government is dealing with different issues of the same topic that are connected, this might lead to seemingly high attention (measured in numbers of hearings, etc.). More prominently, the problem of measurement errors was raised by Dowding, Hindmoor, and Martin (2015). They somehow miss the point by arguing attention would be a "proxy measure" for importance. But they are right to stress that data from PAP is not necessarily measuring what we believe it to be. In his answer to this critique, Jones stresses that PAP.

> tabulates incidences of policy activity by content categories. This activity can certainly involve talking about something - as is the case for speeches by the executive. In other cases, it is incorrect, as is the case for counts of laws [...] or regulations [...]. In the case of parliamentary questions or Congressional hearings, the measure is certainly what government is doing, although it is also a measure of what topics the government is paying attention to (Jones, 2015, p. 39).

"Policy activity by content categories" involves clearly less ambiguity than "attention". So, the aim is, of course, to reduce measurement noise. Nevertheless, in doing so, the modeled process is probably biased and deviations of predictions from observations will not necessarily be purely random.

The second problem, why it is so hard to follow a specific signal through the policy process, is complexity:

> Interactions of a system with its environment are seldom linear and direct. One must appreciate internal system dynamics as well as external inputs to understand system outputs. The adjustment incorporates potentially complex interactions between the internal parts of the system and its environment. These interactions are often governed by simple processes, but they can combine in ways that generate a great deal of complexity (Jones & Baumgartner, 2012).

Compared with this characterization, political science models of the policy process seem to lack complexity in too many ways. Based on formal analysis and comparison with US budget data (Jensen, Mortensen, & Serritzlew, 2015) that models should account for different amounts of friction. This research is focused on the *data generating process*. Deep learning might provide a different solution to the problems of noise and complexity. The presented example gives empirical evidence that the *outputs* of the policy process as an information-processing system where attention leads to stability *and* fundamental policy shifts can be modeled in a neural network. But the potential of real deep learning goes far beyond this basic show-case model. In the following section, I will shortly describe some more advanced features of deep learning that might allow for even better models.

6. Advanced deep learning

A very useful aspect of neural nets has not yet been mentioned: neurons may be very simple, but they can be combined in the most flexible way. For example, there is the possibility to create so called "context layers". Neurons in these layers are storing the values of a hidden layer. The whole system gets a kind of memory this way. *Elman networks* (Elman, 1993) fall into this category. The advantage is that the former status of the system is treated as its own input. With this "trick", developments over time like autocorrelation can be modeled.

For PET this might be very important. First, all stochastic process approaches have tried to get rid of autocorrelation, because the estimation of a PDF is only valid under the assumption that data is not autocorrelated. Therefore, data is transfered to percentage changes. But due to this transformation, parts of the signal might get lost and new sources for noise are introduced. As an example, Jones et al. (2009) refer to budget punctuations in Germany. One of the biggest punctuations that is presented (because the focus is on annual percentage changes) is de facto an increase in the federal budget to support sport events in the early years of the Federal Republic of Germany that is nearly invisible in absolute terms. In addition, input and output of the policy process may lie on different time scales. Hegelich et al. (2015) find that there is a connection between attention and budgets with a time lag of two years, which is very plausible because in the US the first budget plans are presented two years in advance. Inte-

grating context layers in deep learning models might therefore improve the accuracy of the predictions.

The second feature of neural networks worth mentioning is that they are not necessarily limited to one output variable. Fig. 1 demonstrates this idea already. If one of the hidden layers would be defined as output layer, the model is delivering multiple outputs for every input.

For PET this opens the possibility to predict values for all budget functions at once. Due to the fact, that a dollar can only be spent once, this makes a lot of sense. The same is true for attention: if any topic is in the focus of political agents, this will reduce the attention they can give to other issues (see Hegelich et al., 2015, p. 233).

Finally, neural networks can perform *unsupervised learning* tasks as well. This means, the model can find patterns in the data and reduce its dimensions without referring to previous reassured observations. Interestingly, this links deep learning to parametric approaches. The idea is that we have two (or more) layers, where all neurons in one layer are connected to the other layer and vice versa. But no neuron is connected to a neuron in the same layer. When optimized, the second layer will learn the probability function of the first layer (Lewis, 2016, pp. 179–181). But the advantage is, this function is not defined by the scientist but generated from data. Neural networks of this kind are called *Restricted Boltzman Machines* (RBM). A *deep belief network* finally is a combination of stacked RBM. With this trick, deep belief networks can handle extremely complex data structures: the model *learns* which patterns in the data form useful features and how this features in the end can be used for sound predictions. It would be very interesting to see what such a machine learning system could generate from the rich data of the Comparative Agenda Project (CAP).

7. Outlook

While data available for political scientists is steadily growing in an exponential way, the methods to analyze complex systems are not within the common tool box of political scientists. Re-establishing the a previously strong connection of political science and computer science in the tradition of outstanding researchers like Herbert Simon and Paul Werbos and thereby defining *political data science* might be crucial for understanding developments with disruptive changes.

But the increase in complexity in deep learning has three negative effects that should be kept in mind: First, the increase of layers makes models more computational intensive. "With N observations, p predictors, M hidden units and L training epochs, a neural network fit typically requires $O(NpML)$ operations" (Friedman et al., 2001, p. 414). But computational power does not seem to set a real limit to applications in political science. There are clever algorithms to reduce the number of operations, e.g. by defining "forget gates" in *Long Short-Term Memory networks* (Gers, Schmidhuber, & Cummins, 2000, p. 2451). In addition, specialized hardware is continuously developed that allows for faster execution of deep learning on parallelized GPUs. Second, neural networks can produce quite different results based on different starting values and random weights. In practice, neural networks are fitted many times and finding the right parameters is more an art than science. A state of the art cross-validation approach is therefore necessary to avoid misleading results. The third problem is more severe: Deep learning is not simulating the policy process but is fitting a function that produces similar outputs. Therefore, it is very difficult to gain deeper theoretical insides about the policy process from these models. The advantage of deep learning for political science will ly in sound predictions but not necessarily in better theories. Although, better predictions might not only help politicians but also guide political scientist to new theoretical models in which the variables and effects found in deep learning models are integrated.

Appendix A. Additional tables and data

See Table A.3.

Table A.3
Budget codes and refering PAP codes.

	BudgetCode	BudgetTopic	PAPCode	PAPTopic
1	50	National Defense	16	Defense
2	150	International Affairs	19	International Affairs and Foreign Aid
3	250	General Science, Space, and Technology		
4	270	Energy	8	Energy
5	300	Natural Resources and Environment	7	Environment
6	350	Agriculture	4	Agriculture
7	370	Commerce and Housing Credit		
8	400	Transportation	10	Transportation
9	450	Community and Regional Development		
10	500	Education, Training, Employment, and Social Services	6	Education
11	550	Health	3	Health
12	570	Medicare		

(*continued on next page*)

Table A3 (*continued*)

	BudgetCode	BudgetTopic	PAPCode	PAPTopic
13	600	Income Security		
14	650	Social Security	13	Social Welfare
15	700	Veterans Benefits and Services		
16	750	Administration of Justice	12	Law, Crime, and Family Issues
17	800	General Government	20	Government Operations
18	900	Net Interest		
19	920	Allowances		
20	950	Undistributed Offsetting Receipts		
21			1	Macroeconomics
22			2	Civil Rights, Minority Issues, and Civil Liberties
23			5	Labor and Employment
24			9	Immigration
25			14	Community Development and Housing Issues
26			15	Banking, Finance, and Domestic Commerce
27			17	Space, Science, Technology and Communications
28			18	Foreign Trade
29			21	Public Lands and Water Management

References

Baumgartner, F. R., & Epp, D. A. (2013). Explaining punctuations. In *Annual meeting of the comparative agendas project, Antwerpen*.

Berk, R. A. (2006). An introduction to ensemble methods for data analysis. *Sociological Methods & Research, 34*(3), 263–295.

Box, G. E. (1979). Robustness in the strategy of scientific model building. *Robustness in Statistics, 1*, 201–236.

Breunig, C., & Jones, B. D. (2011). Stochastic process methods with an application to budgetary data. *Political Analysis, 19*(1), 103–117.

Breunig, C., & Koski, C. (2012). The tortoise or the hare? incrementalism, punctuations, and their consequences. *Policy Studies Journal, 40*(1), 45–68.

Dowding, K., Hindmoor, A., & Martin, A. (2015). The comparative policy agendas project: Theory, measurement and findings. *Journal of Public Policy*, 1–23.

Elman, J. L. (1993). Learning and development in neural networks. *Cognition, 48*(1), 71–99.

Epp, D. A., & Baumgartner, F. R. (2016). Complexity, capacity, and budget punctuations. *Policy Studies Journal*.

Flink, C. M. (2015). Rethinking punctuated equilibrium theory: A public administration approach to budgetary changes. *Policy Studies Journal*.

Friedman, J., Hastie, T., & Tibshirani, R. (2001). *The elements of statistical learning. Springer series in statistics* (Vol. 1). Berlin: Springer.

Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with lstm. *Neural Computation, 12*(10), 2451–2471.

Harvey, A. C., & Todd, P. (1983). Forecasting economic time series with structural and box-jenkins models: A case study. *Journal of Business & Economic Statistics, 1*(4), 299–307.

Hegelich, S. (2016). Decision trees and random forests: Machine learning techniques to classify rare events. *European Policy Analysis (EPA), 2*(1), 98–120.

Hegelich, S., Fraune, C., & Knollmann, D. (2015). Point predictions and the punctuated equilibrium theory: A data mining approach – Us nuclear policy as proof of concept. *Policy Studies Journal, 43*(2), 228–256.

Howlett, M. (2009). Process sequencing policy dynamics: Beyond homeostasis and path dependency. *Journal of Public Policy, 29*(03), 241–262.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (Vol. 112). Springer.

Jensen, J. L., Mortensen, P. B., & Serritzlew, S. (2015). The dynamic model of choice for public policy reconsidered: A formal analysis with an application to us budget data. *Journal of Public Administration Research and Theory*, muv007.

John, P., & Bevan, S. (2012). What are policy punctuations? large changes in the legislative agenda of the uk government, 1911–2008. *Policy Studies Journal, 40*(1), 89–108.

Jones, B. D. (2003). Bounded rationality and political science: Lessons from public administration and public policy. *Journal of Public Administration Research and Theory, 13*(4), 395–412.

Jones, B. D. (2015). The comparative policy agendas projects as measurement systems: response to dowding, hindmoor and martin. *Journal of Public Policy*, 29–44.

Jones, B. D., & Baumgartner, F. R. (2005). *The politics of attention: How government prioritizes problems*. University of Chicago Press.

Jones, B. D., & Baumgartner, F. R. (2012). From there to here: Punctuated equilibrium to the general punctuation thesis to a theory of government information processing. *Policy Studies Journal, 40*(1), 1–20.

Jones, B. D., Baumgartner, F. R., Breunig, C., Wlezien, C., Soroka, S., Foucault, M., François, A., Green-Pedersen, C., Koski, C., John, P., et al. (2009). A general empirical law of public budgets: A comparative analysis. *American Journal of Political Science, 53*(4), 855–873.

Jones, B. D., Baumgartner, F. R., & True, J. L. (1998). Policy punctuations: Us budget authority, 1947–1995. *The Journal of Politics, 60*(01), 1–33.

Jones, B. D., Sulkin, T., & Larsen, H. A. (2003). Policy punctuations in american political institutions. *American Political Science Review, 97*(01), 151–169.

Jones, B. D., Zalányi, L., & Érdi, P. (2014). An integrated theory of budgetary politics and some empirical tests: The us national budget, 1791–2010. *American Journal of Political Science, 58*(3), 561–578.

Langley, P., & Simon, H. A. (1995). Applications of machine learning and rule induction. *Communications of the ACM, 38*(11), 54–64.

Lewis, N. (2016). *Deep learning made easy with R: A gentle introduction for data science*. CreateSpace Independent Publishing Platform.

March, J. G., & Simon, H. A. (1958). *Organizations*. Wiley.

Simon, H. (2000). Public administration in today's world of organizations and markets. *PS: Political Science & Politics, 33*(04), 749–756.

Simon, H. A. (1955). A behavioral model of rational choice. *The Quarterly Journal of Economics*, 99–118.

Thomas III, H. F. (2016). Contagion in policy agendas: Inferences from an agent-based model of issue attention. In *MPSA annual conference*.

Werbos, P. J. (1994). *The roots of backpropagation: From ordered derivatives to neural networks and political forecasting* (Vol. 1). John Wiley & Sons.