THE UNIVERSITY OF CHICAGO

Masters in Computational Social Science

Reid Harry McIlroy-Young
June, 2018

,

Faculty Advisor: Dr. JAMES A. EVANS
Preceptor: JOSHUA MAUSOLF

A paper submitted in partial fulfillment of the requirements for the Master of Arts degree in the Master of Arts in Computational Social Science

Contents

List of Figures

List of Tables

1 Introduction

The mass proliferation of micro computers that began last century has had, and continues to have an incredible impact on society (Weizenbaum, 1972), the economy (Gordon, 2000) and information distribution (Berners-Lee et al., 2010). The access to computers – the scale, quality and depth of data they provide – has also had major effects on science (Lazer et al., 2009); but analysis has been primarily focused on either the macro cultural effects (Pfaffenberger, 1988) or the economic usage (Landauer, 1995) of computers.

By comparison, the usage of computers by scientists has been overlooked by researchers. The impact has spread, at different rates, throughout all the domains of science. The social sciences have been one of the last fields to be significantly affected, but this situation is likely to change this century (Watts, 2007). There are large datasets of medical, social and economic data generated by a complex web of automated systems whose implications we are not yet able to interpret at more than a surface level (Kossinets and Watts, 2006) (Back et al., 2010), with many examples of even these surface level analyses failing (Lazer et al., 2014) (Kramer et al., 2014). One aspect of this ‘paradigm shift’ that has received much less scrutiny is how the individual scientists are interacting with new computational tools. Are social scientists still able to do solid research with traditional with the traditional pre computational techniques or, or have computers changed?

One of the main methods for large scale analysis of the culture or structure of scientific work involves bibliometric techniques (De Bellis, 2009) using large standard datasets tracing citations between many articles (e.g. Boyack et al., 2005; Börner, 2010, 2015; Sugimoto et al., 2013; Shi et al., 2015; Evans and Foster, 2011;

Skupin et al., 2013). These datasets have generally lacked information about the computational aspects of the works they contain. For example the Clarivate Analytics Web of Science (WOS) does not have any structured metadata on the computational methods used by the publications in the database, such as the source code (McLevey and McIlroy-Young, 2016), and as such, research into this dimension is non-trivial. Nevertheless, recent developments in natural language processing (NLP) have shown that complex concepts can often be extracted reliably from text, providing a pathway to conducting research on computational methods in the literature (Evans and Aceves, 2016) (Foster et al., 2015).

Large scale surveys of social scientists to determine computational usage have not been done, although much has been hypothesized regarding how computers or other computationally derived methods will revolutionize science (e.g. De Jong and Rip, 1997; Anderson, 2008; Provost and Fawcett, 2013; John Walker, 2014). Moreover, even quantitative surveys of the literature have not been done. We can infer from the proliferation of new computational journals, such as the *Journal of Computational Social Science*, the first issue of which published in January 2018, that there is a demand for computational knowledge and techniques within some domains of social science,although its extent and means of spreading is not well characterized. This paper aims to begin to correct this lack of solid knowledge regarding computational approaches in social science. Once the usages of computation can be identified, there is a large number of existing frameworks from which the diffusion (Griliches, 1960), spread (Padgett and Ansell, 1993) and innovation (Foster et al., 2015) of computational methods can be viewed.

In this paper, I am using a bidirectional Long Short Term Memory (LSTM) based classifier Graves and Schmidhuber (2005), which that I have been developing, for extract useful classifications with only partially or loosely classified data. The specific task is to identify the description of computational style, techniques, master, and approaches in journal articles, talks and other publications outside of the explicitly computational publications, conferences, or other sources. This paper seeks to both quantify the level of computational usage across publications in the social sciences and consider their implications.

2 Methodology

2.1 Data

The source of data used for this analysis is the Web of Science (WOS) database hosted by *Knowledge Lab*, a research group at the University of Chicago focused on developing a science of science (Chu and Evans, 2018) with large-scale publication resources. The Web of Science is owned by *Thompson Reuters* until 2016, when it transferred to *Clarivate Analytics*, who now maintains it (Clarivate Analytics, 2016). Our subset contains metadata from the most cited scientific and social scientific publications from 1960 to 2015, with newer records being more complete. The data used are a subset of this larger collection, more specifically, the 1,457,418 publications with a Social Science subject classification from 2005 to 2015. The classifications of the publications are derived from the journals or conferences (hereafter called source) from which publications are drawn (Knowledge Lab, 2017). These classifications were grouped into six broad subjects (e.g. Natural Sciences, Social Sciences), each of which contain multiple subject categories (e.g. Cultural Studies or Ergonomics). Each source has up to three subject categories, which are assigned manually by WOS annotators and there could be many combinations of subject classes that map to the same broad topic (Efremenkova and Gonnova, 2016). The classification of each source can then be propagated to their publications, and thus I can assign a publication to one or more disciplines and sub-disciplines within a well defined hierarchy (Thomson Reuters, 2012). A summary of the data is shown in Table 9. Appendix A contains the complete list of tags used along with their subjects.

To derive information about the computational nature of soical science publications, the 11,115 sources from 2005 to 2017 containing explicitly computational subjects in the 'Computer and information sciences' sub-discipline were intersected with the journals in the social sciences, yeilding 782 sources containing 106,680 social science publications that are explicitly computational in nature. Table 2 provides the breakdown according to subject, where the percentage of explicitly computational works notably varies by a massive factor of 50, from 0.77% in Law to 38% in Media and Communication.

In addition to the subject tags, the paper abstracts, titles, authors, publication month, publication year, and unique identification number (*WOS ID*) were also gathered and used later in the analysis.

I also employed an additional dataset to help disaggregate computational features from WOS publications, the Stack Exchange Data Dump. This is a collection of user created technical question-answer data from the Stack Exchange network, hosted by the Internet Archive under a creative commons attribution share alike 3.0 license (Stack Exchange network, 2018). The Stack Exchange network is a large collection of question and answer websites divided by category. In this case, the Stack Overflow (for general programming questions), Economics and Psychology & Neuroscience Stack Exchange sites are of interest.From the Stack Exchange data dump, I used the user generated tags – between one and three – assigned by highly ranked community members.

	Number of Publications	Number of Sources	Number of Subjects	Example Subject Categories
Economics and Business	459,992	2363	6	Business
Psychology	366,313	1017	11	Ergonomics
Educational Sciences	168,342	1009	3	Education, Special
Sociology	136,894	777	9	Sociology
Political Science	95,920	609	3	Political Science
Other Social Sciences	83,611	571	4	Asian Studies
Media and Communication	63,625	518	2	Communication
Law	40,829	295	2	Law
Full Data set	1,457,418	6893	46	

Table 1: *Summary of the subjects of WOS data. Note that publications can have multiple subjects, thus the final row is not a sum of the rows above.*

	Computational Publications	Percentage Explicitly Computational	Example of Explicitly Computational Source
Economics And Business	60,602	13.17	Decision Support Systems
Psychology	5364	1.46	Interacting With Computers
Educational Sciences	17,988	10.69	Computers & Education
Sociology	3975	2.90	Persuasive Technology
Political Science	1815	1.89	Electronic Government
Other Social Sciences	2829	3.38	Adaptive Behavior
Media And Communication	24,798	38.98	Scientometrics
Law	313	0.77	Law And The Semantic Web
Full Data Set	106,680	7.32	

Table 2: *Distribution of explicitly computational publications, note publications can have multiple subjects, thus the final row is not a sum*

2.2 Preprocessing

In the WOS data, the publication year and month fields are provided as integers, so they required no preprocessing. Meanwhile, the other fields are all UTF-8 encoded raw texts, and I first tokenized both the title and the abstract by separating the words into separate strings. Then I used these texts to word embeddings. The tokenizing was done with the *Natural Language Toolkit*'s (*NLTK*) (Bird, 2006) English language tokenizers, and more specifically, sentence tokenizing was done with the Punkt system (Kiss and Strunk, 2006) and the words tokenized with a collection of regular expressions derived from the Penn Treebank (Marcus et al., 1993). The other fields were treated as raw texts, without any additional transformations.

The final step in preparing the data for the neural network was constructing the word embeddings. A word embedding is a mapping from a collection of words to a collections of vectors in a high dimensional vector space with similar words closer together in the space (Jurafsky and Martin, 2014). I used the *Word2Vec* method (Mikolov et al., 2013) implemented in *gensim* (Řehůřek and Sojka, 2010). *Word2Vec* gives word vectors whose dimensions can be interpreted as a collection of semantic directions. For example, one direction could be the masculine-feminine dimension of the word while another could be the 'computationalness' of the word (Bolukbasi et al., 2016). This relationship is derives from the neural auto-encoder used to perform the embedding as an approximation of SVD matrix factorization with words as rows and word contexts or windows as columns (Levy and Goldberg, 2014) that minimizes the point-wise mutual information, thus the space has many of the properties of a Euclidean Hilbert space. Unfortunately, determining the meaning of any one derived dimensions is impossible in practice, due to their number (in this case, 200 dimensions) and their non-linear relationships (the masculine-feminine direction, if it exists in the particular embedding, is a weighted combination of all 200 basis directions and is likely not orthogonal to 'computationalness'). However, the embedding is still very useful as it assigns similar words with similar vectors and makes their divergences depend on their difference in meanings, e.g. king and queen diverge in similar ways to man and woman. The word embeddings act as a foundation for the deep neural network to build on and are the standard first step (Goodfellow et al., 2016a) as it converts variable length strings into fixed length vectors. For all embeddings, hierarchical softmax was used with a window size of five, 200 dimensional output, five iterations and no removal of words for any reason. This configuration should allow for infrequent words to be embedded nearly optimally (Řehůřek and Sojka, 2010), but there is also likely room for improvement.

An alternative to an auto-encoder is a one-hot vector representation where each word is a dimension of the input vector and all but one of the values is zero, with the non-zero element (usually equal to one) corresponding to the selected word's dimension. This method greatly increases the size of the first layer as in this case there are 11,430,321 unique words in the combined corpus, and it means that the RNN (Recurrent Neural Network) has to learn the meaning of each word separately as their location in the vector space provides no additional information. Thus, these methods are most commonly used for much simpler inputs, e.g. character level tasks (Rodriguez et al., 1999).

2.3 Model Selection

Before undertaking the usage of a complex model such as that outlined above, it is worth considering the alternatives. There are many other ways to classify texts. A Naive Bayes model would generally be the first choice (McCallum et al., 1998), while a K-nearest, SVM, ensemble or logit methods could also work. Unfortunately, none of these models were able to obtain anything better than random guessing, and most of them performed worse than random guessing when a pilot study was performed. The Naive Bayes model was the best performer, arriving at nearly 95% accuracy on the training data, but when applied to the holdout set, it was worse than random guessing. This problem of over-fitting or for some models, always guessing negative, was present in all these models. Thus a model with a more nuanced understanding of language, such as a neural network, is required.

2.4 Model Description

The classification of the publications was done with two separate, two layer bidirectional (Graves et al., 2013) Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) Recurrent Neural Networks (RNNs), with the LSTM implementation used is *NVIDIA*'s *cuDNN* (Chetlur et al., 2014), see Figure 1 for an illustration. LSTM cells are a variant of RNN cells, which work by taking in two vectors, the input vector (in this case, the word2vec vector of a word) and the recurrent vector which is initially all zeros. These two vectors are appended together to form a single input which is fed to a single layer neural network whose output layer has two components, the required output (in this case, the classification) and an output recurrent vector having the same size as the input recurrent vector.

The output recurrent vector is then fed to the cell along with the next item in the sequence, thus giving a stateful component to the next output. Unfortunately, the memory of the cell tends to have short life span (Greff et al., 2017) as every new input is combined with the memory vector. The LSTM architecture solves this by adding a second output vector, the blue lines in Figure 1. This second vector acts indirectly on the input and is in turn indirectly modified, via the peepholes, by the LSTM cell in each cycle. This configuration means that the *long term* memory vector is modified much less by each input and thus retains more of the history. The weights of the cell are trained by backpropagation(PyTorch core team, 2017) just like most other neural network architectures, and the specifics of which have been thoroughly explored(Riedmiller and Braun, 1993).

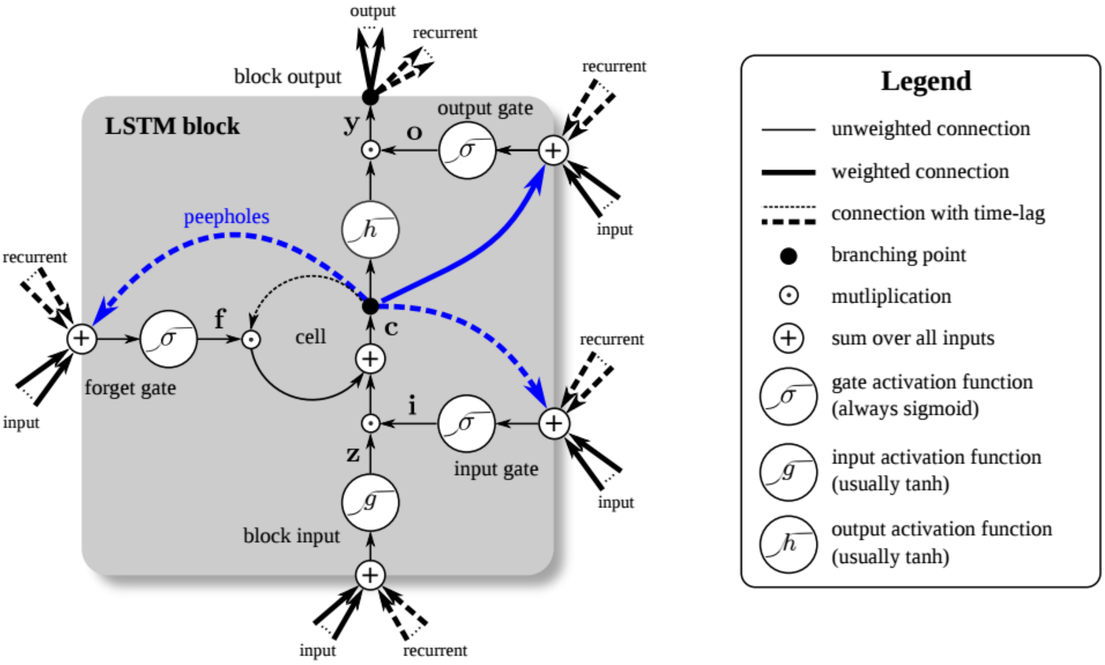


Figure 1: *Illustration of LSTM cell as implemented by NVIDIA*
Image by Klaus Greff and colleagues as published in LSTM: A Search Space Odyssey (Greff et al., 2017)

To create the classifier used in this paper, 16 LSTM cells were used, their arrangement is shown in Figure 2, which shows the network's structure and the flow of data through the model. Starting from the bottom, the abstract and the title are provided as raw text, which then go through the two preprocessing steps, tokenizing and embedding, and then the result of which is that each word becomes a vector $\vec{x}_t = \begin{bmatrix} 0 & .2 & 0 & \dots & .3 & 0 \end{bmatrix}^T$. The process is identical, modulus weights, for the title and abstract, and for this example we will consider only the title. A word's vector is fed to the first forward layer (h_t^1) of the RNN, which applies the non-linear transforms described above, before finally outputting three 128 dimensional vectors (*long term*, *short term* and an output vector). The output is then fed to the next layer ($h_t^1 \rightarrow h_t^2$) while the memory vectors are given back to the cell along with the next word vector ($h_t^1 \rightarrow h_{t+1}^1$). The next layer repeats the process with the output ignored, until the final word is provided. These ignored outputs can be of use and will be discussed. Simultaneously, the last word is provided to the first reverse layer (g_t^1) where the same process as the forward layer takes place, the output is fed up the network ($g_t^1 \rightarrow g_t^2$) while the memory vectors go to the next (previous word in the text) word ($g_t^1 \rightarrow g_{t+1}^2$). The final output vectors (128 dimensional) from the final word in the title in the forward direction along with the final word from the backward direction are appended to each other to create one vector ($\oplus$), a 256 dimensional vector giving the module's 'thoughts' on what was in the title. This vector is then combined with the output from the abstract and given to a final neural network (u) that looks at the two combined 512 dimensional vector and produces a single 2 dimensional vector ($\vec{y}$). The two dimensions provided are the log-odds of the publication *not* being computational and the log-odds of the publication being computational.

When the neural network first runs on a publication, its weights, the parameters for the non-linear transforms, are randomly selected, so it's outputs are useless. Thus it has to be trained. To train the model, a training dataset is created and labelled with the correct outputs for each of the publications in the set. Then publications are randomly selected from the set to be fed through the network and to have the outputs compared to the expected outputs. Through the use of backpropagation (Werbos, 1982) (differentiating the output with respect to every parameter and every input), the given output and correct output are compared and the network's weights are updated to move future outputs more in line with the correct outputs. The networks are then tested for accuracy against a separate set of publications, the testing set, once an epoch (in this case, 500 training examples), to see if it is making accurate predictions.

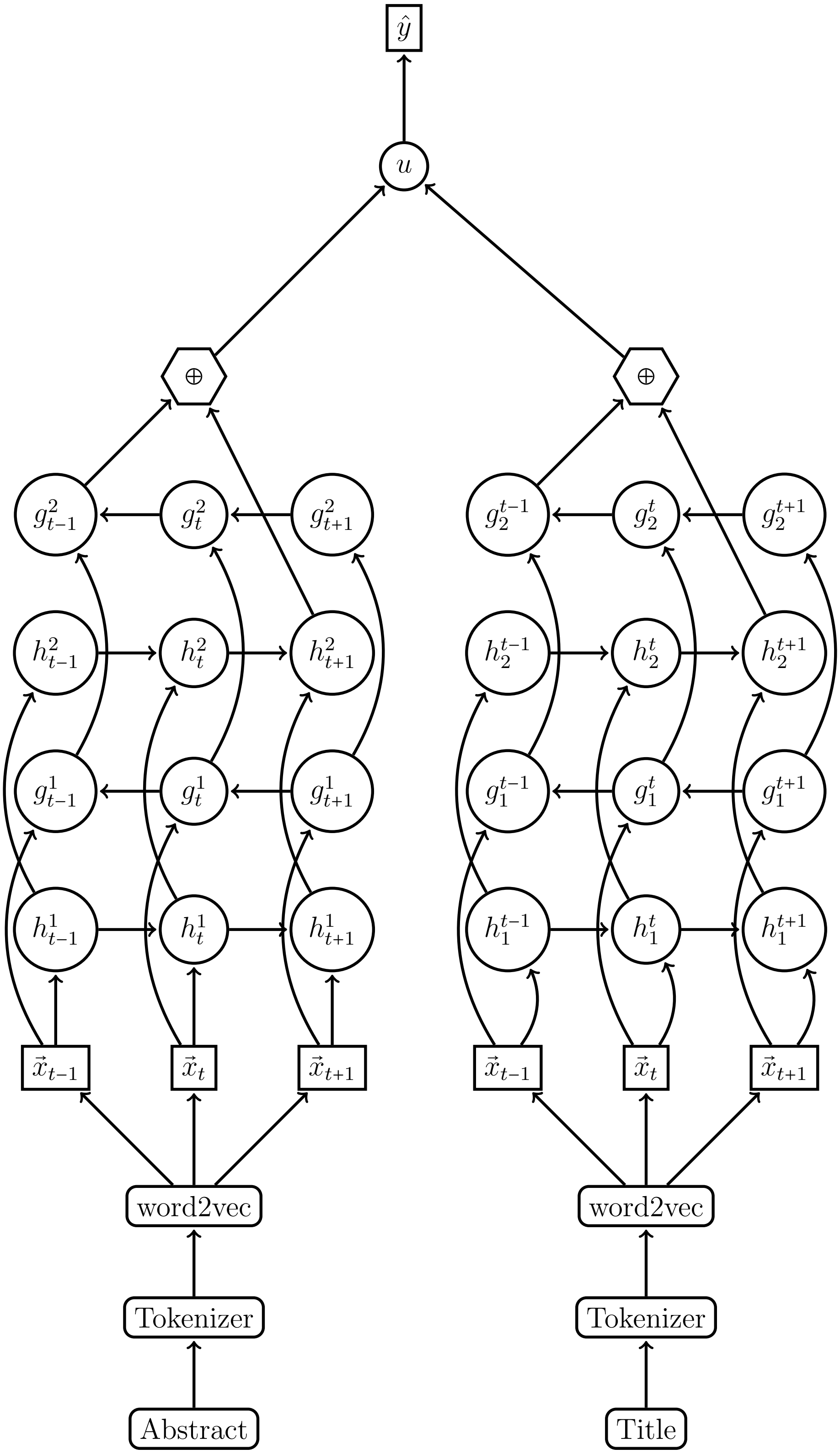

$\hat{y}$
u
$\oplus$
$\oplus$
g^2_{t-1} g^2_t g^2_{t+1}
g^{t-1}_2 g^t_2 g^{t+1}_2
h^2_{t-1} h^2_t h^2_{t+1}
h^{t-1}_2 h^t_2 h^{t+1}_2
g^1_{t-1} g^1_t g^1_{t+1}
g^{t-1}_1 g^t_1 g^{t+1}_1
h^1_{t-1} h^1_t h^1_{t+1}
h^{t-1}_1 h^t_1 h^{t+1}_1
$\vec{x}_{t-1}$ $\vec{x}_t$ $\vec{x}_{t+1}$
$\vec{x}_{t-1}$ $\vec{x}_t$ $\vec{x}_{t+1}$
word2vec
word2vec
Tokenizer
Tokenizer
Abstract
Title

Figure 2: *Simplified recursive bidirectional Recursive NN (RNN) layout, LSTM connections were removed for clarity. Circles are NN layers, rectangles with curved corners are the preprocessing, hexagons are combined outputs from the RNN layers, and final output and inputs are indicated with squared corners.*

3 Results

3.1 Training

Two separate approaches were attempted for training the network, namely, using one network per subject (e.g., media & communications, economics & business, sociology) and using one network for the complete dataset. Training the network proved to be more difficult than expected. Data gathering took many hours. Moreover, limitations of the database made bibliographic data collection difficult, and data prepossessing took approximately two weeks on a high-end laptop.

3.1.1 One Network Per Subject

To reduce the word embedding run time, each subject was separated and the embeddings run independently, with *Law* taking a few hours and *Economics and Business* multiple days to complete. Since the embeddings are independent across subjects, each subject has to have a separate network trained for it.

To train the networks, all the explicitly computational records were combined with a random sample of non-explicit records of twice the size of the explicit set, except in *Media And Communication* where the complete collection was used due to proliferation of computational journals. Then a holdout set of 10% was removed to be used for cross validation.

The models were trained with stochastic gradient descent with the Adam optimizations (Collins et al., 2012) and a cross entropy loss. Equation 1 gives the exact formulation used (PyTorch core team, 2017). During training, after every 500 updates to the model (1 epoch), the testing set error was checked. The preferred way to do this is with a loss function, in this case, the loss (Goodfellow et al., 2016b) was the cross-entropy of the expected output and the actual output . The cross-entropy can take any positive value, although less than 1 is expected after training. Using average loss to quantify the model's accuracy instead of the raw error rate is preferable as it is less jittery (Goodfellow et al., 2016b) and picks up small improvements (or deteriorations) missed with a binary successful/not successful. There are many other measures available, such as the area under the testing receiver operating characteristic curve or the F_1 score (James et al., 2013), but those are more useful for tuning the model and many require additional computation.

$$loss(\boldsymbol{output}, \boldsymbol{expected}) = -\log\left(\frac{\exp(\boldsymbol{output} \cdot \boldsymbol{expected})}{\sum_j \exp(\boldsymbol{output}_j)}\right) \tag{1}$$

	Testing Error	Loss
Economics And Business	0.25	0.46
Psychology	0.12	0.27
Educational Sciences	0.17	0.37
Sociology	0.10	0.26
Political Science	0.15	0.46
Other Social Sciences	0.16	0.34
Media And Communication	0.24	0.49
Law	0.09	0.92

Table 3: *The testing loss and error for each subject on the epoch used for the analysis*

Figure 3 shows the loss on the testing set across the epochs for each of subject's models. Notice the slow drop of 0.10 or so from epoch 0 to epoch 30, at which point they cease decreasing. The plot for *Law*, is not representative of the model's accuracy and the jump is due more to the small testing set size (94) as a couple

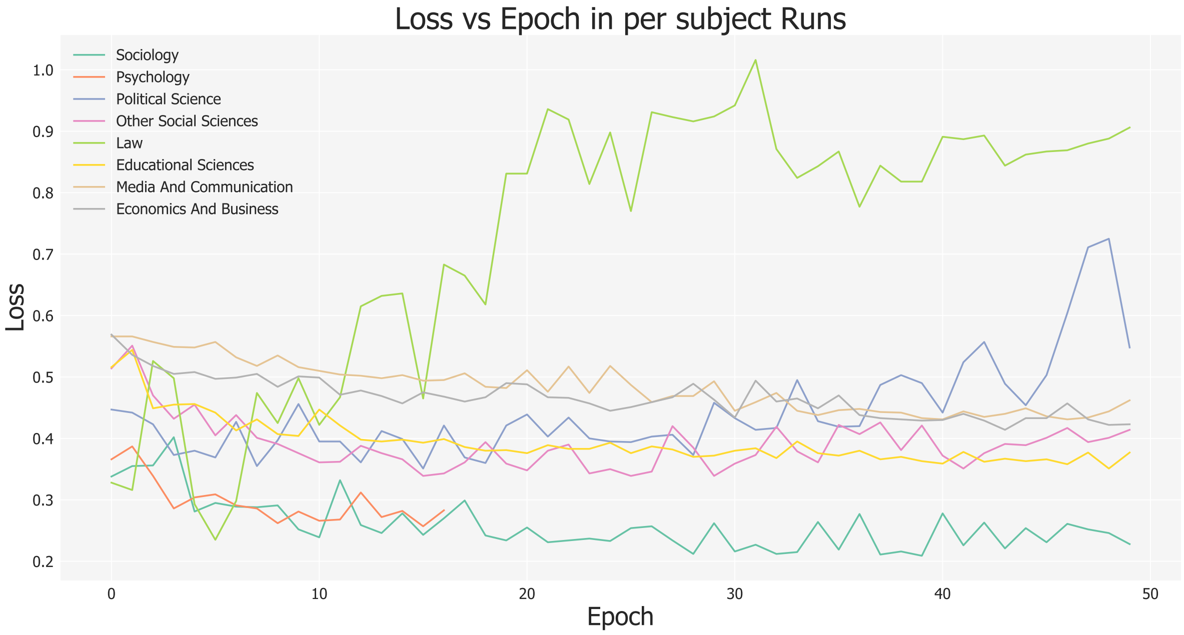


Figure 3: *Testing loss for all subject's models, across all epochs (1 epoch is 500 training exposures), note that some data for Psychology was lost*

of examples being misclassified will cause a large jump. The loss shown for law, is not the first attempt. In the first run, it had the lowest loss and a testing error of 0 for most epochs, which is a sign of over-fitting more than of a quality model, so it was discarded. Over-fitting is a major concern when training neural networks (James et al., 2013) and the characteristic switch from nearly flat slope to positive slope is visible in some of the losses, most notably *Political Science* and *Other Social Sciences*.

To help avoid over-fitting, the testing error rates are also considered, with Figure 4 showing them against epoch number. They are nosier than the loss, so they are a secondary consideration from that discussed above. In the testing error, the errors tend to stabilize near epoch 30, and so epoch 30 was chosen to use for

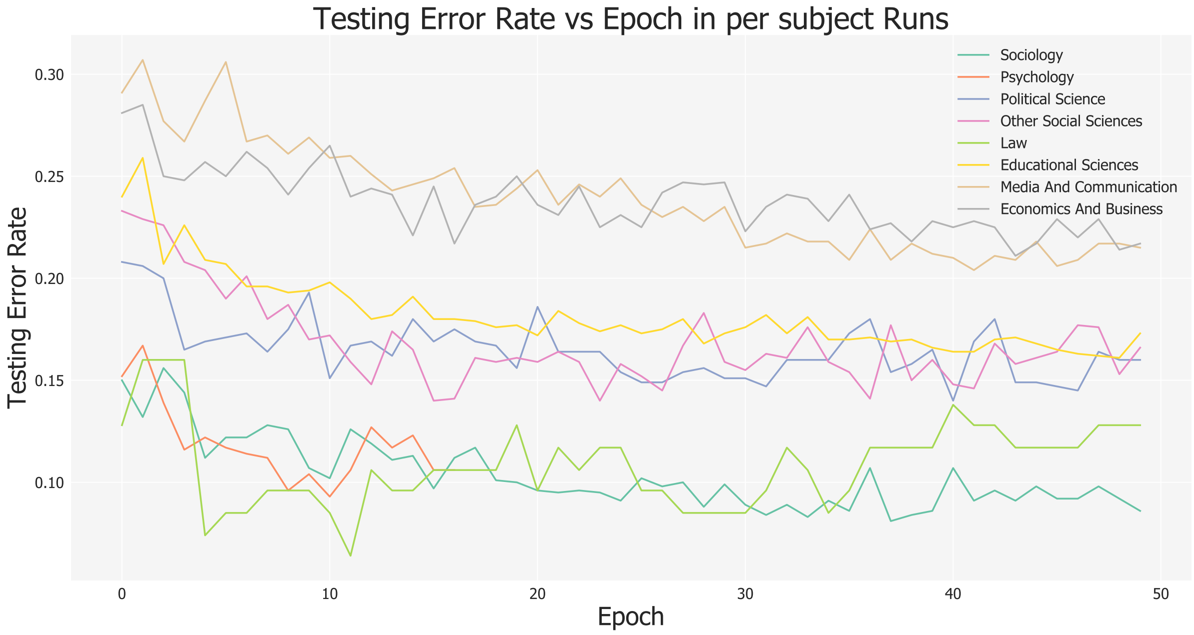


Figure 4: *Testing error for all subject's models, across all epochs (1 epoch is 500 training exposures), note that some data for Psychology was lost*

analyzing the complete dataset. Figure 3 shows the final values of testing loss and testing error at this epoch. Also note that the error rate for *Law* is quite low despite the high loss, which confirms the small dataset is causing anomalies in the loss.

3.1.2 Complete

Once the per subject networks had been trained and tested, training a model on the complete dataset could follow. A word embedding was created across the whole dataset, which took about a week running on a server. Then the explicitly computational records were identified and combined with twice their count of non-computational records and a 10% testing holdout set was created. The model was initially tested on a small subset of the testing data, but at epoch 112, the test set was switched to random samples from the complete set to make sure over fitting was not occurring. The epoch size for this model is 2000 records as it took much longer to converge, and Adam based stochastic gradient descent was still used along with cross-entropy loss.

Figure 5 shows the loss per epoch of the complete model. The model seems to have stopped improving around epoch 250, which is 35,000 more exposures than any of the simple subject networks, thus this was a much slower process taking days on a NVIDIA GeForce GT 750M with 2048 MB of RAM. The final model used for this analysis was from epoch 260 as that had particularity good testing error of 0.13 and loss of 0.32. Compared to any of the per subject networks it is better than all but *Sociology* and *Law*. Of note is that there were 288,646 records in the training set, while the model was exposed to 520,000 records so the model saw each example on average only twice.

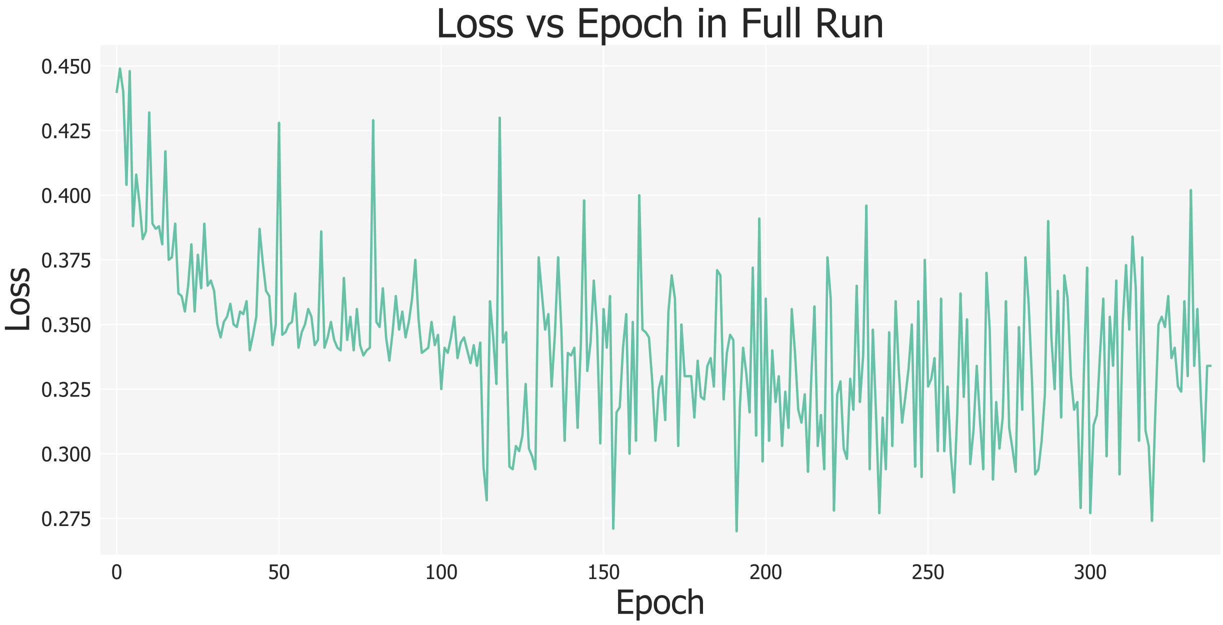


Figure 5: *Testing loss for all complete model, across all epochs (1 epoch is 2000 training exposures), note that at epoch 112 the testing set was expanded*

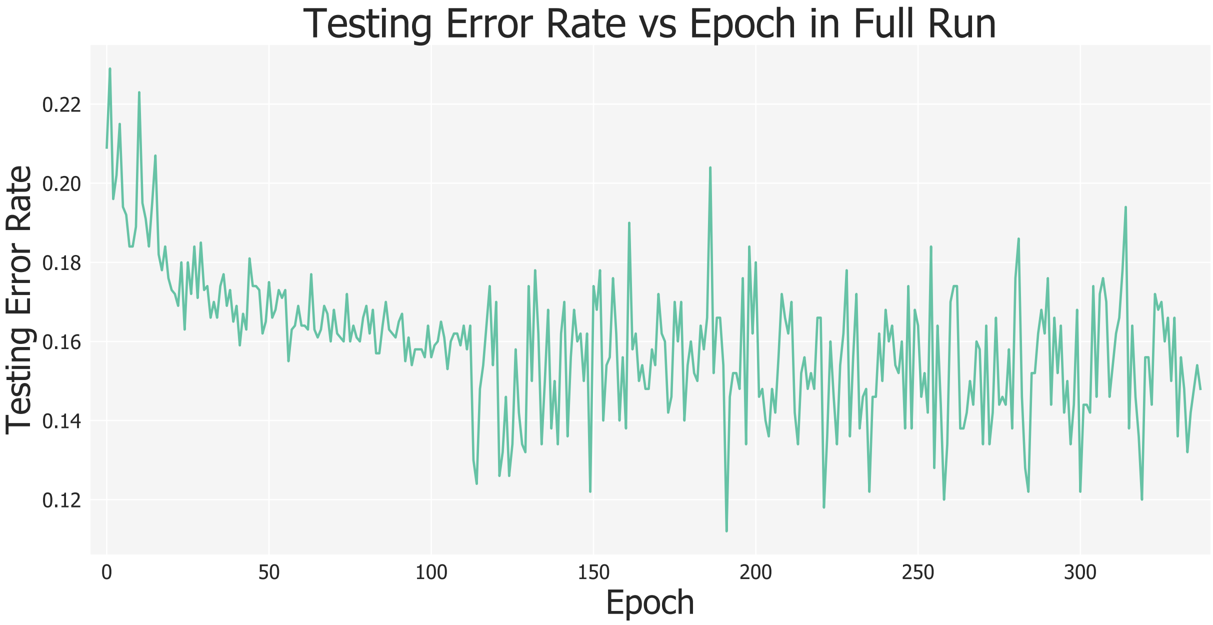
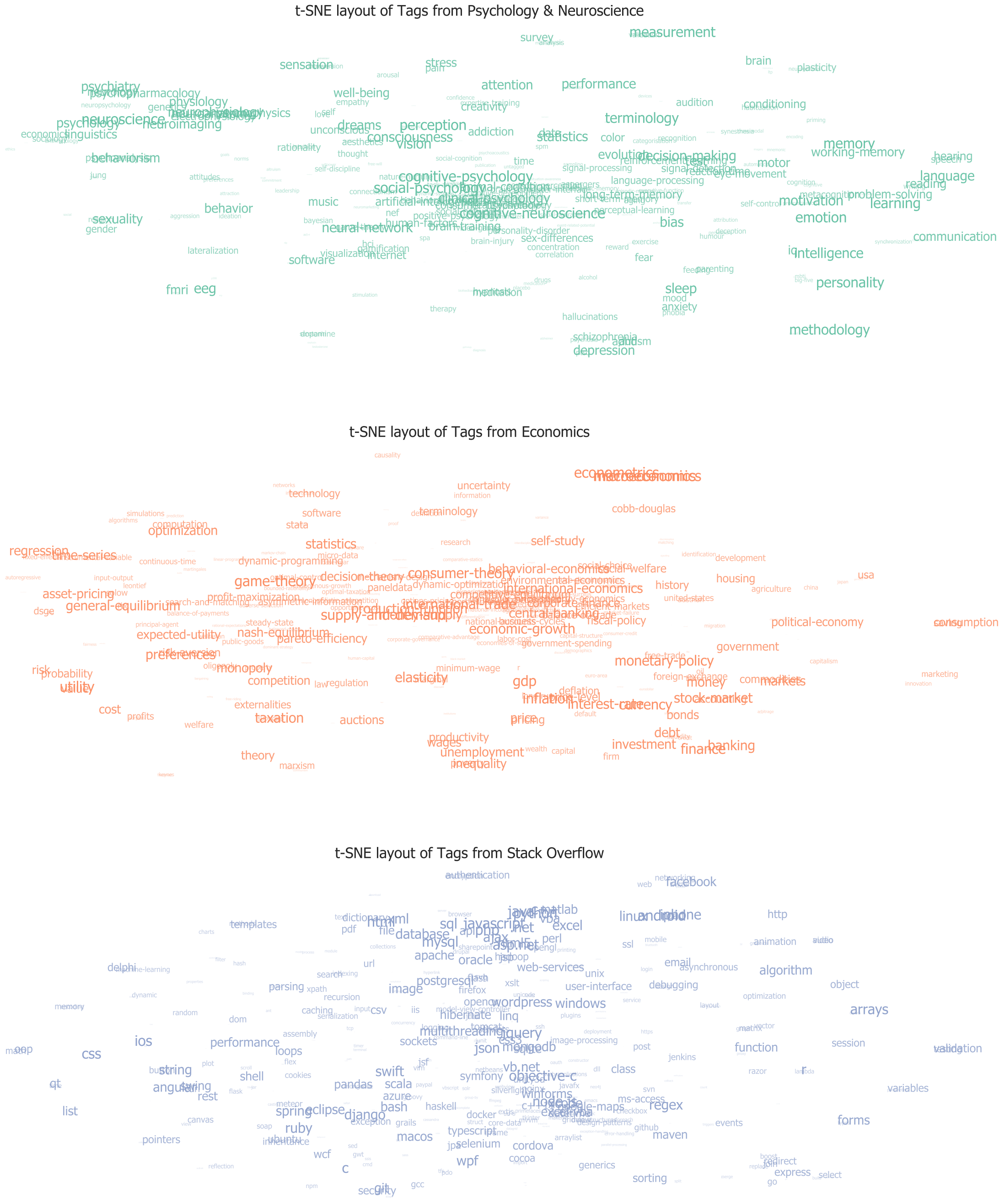


Figure 6: *Testing error rate for all complete model, across all epochs (1 epoch is 2000 training exposures), note that at epoch 112 the testing set was expanded*

3.2 Word2Vec

Before considering the results of the models, the validity of the foundations need to be tested. In particular the Word2Vec embedding space is worth some consideration. Using the tags from stack overflow, a vocabulary of key words can be constructed. Stack Overflow can be considered to be computational, and Economics and Psychology & Neuroscience are two sets of keywords for non-computational subjects. There is a large bias in Stack Exchange towards computational topics as Stack Overflow was the first and largest site. Thus tags for Stack Overflow were trimmed to only those with over 10,000 usages, while the other two sites are unfiltered. Then the tags found in the the Word2Vec embedding space were collected and their embeddings derived, the number of tags used are given in Table 4.

To visualize the distribution of tags, a dimension reduction is required. Principal component analysis was used to reduce the number of dimensions from 200 to 50, and then t-distributed stochastic neighbor embedding (t-SNE) (Maaten and Hinton, 2008) was used to go from 50 to 2. This was done separately for each of the three site's tags, Figure 7 and together, Figure 8, Appendix C has larger versions of the individual plots. Upon examination, the tags are roughly clustered as expected, e.g. `mysql` is near `database` and `gdp` is near `inflation`. But when examined further, there are a few oddities. `C` and `R` are both outliers and are not near other languages. Table 5 gives the nearested neighbours in the Word2Vec space of a few words. Notice that 'C' is near some random strings while 'Python' is near other programming languages. This is a byproduct of the tokenization as 'C' and 'R' are likely incorrectly (or correctly) being separated from equations or initialisms and thus those are the dominant environment, also 'C', in particular, is part of the copyright symbol. The table shows that other words such as 'Sociology' are being correctly placed and thus the embedding as a whole is mostly valid. An interactive and more in depth analysis of the word2vec space is hosted as a in interactive webapp at http://shiny.reidmcy.com/int/.


t-SNE layout of Tags from Psychology & Neuroscience

t-SNE layout of Tags from Economics

t-SNE layout of Tags from Stack Overflow

Figure 7: t-SNE reduction of tags from Stack Overflow, Economics and Psychology & Neuroscience, sized by number of usages on the site

Site	Total Tags	Used Tags	Top Tags
Stack Overflow	51671	374	`javascript,java,php`
Economics	372	275	`macroeconomics, microeconomics,econometrics`
Psychology & Neuroscience	348	296	`cognitive-psychology, social-psychology, neuroscience`

Table 4: *Stack exchange sites tag usage*

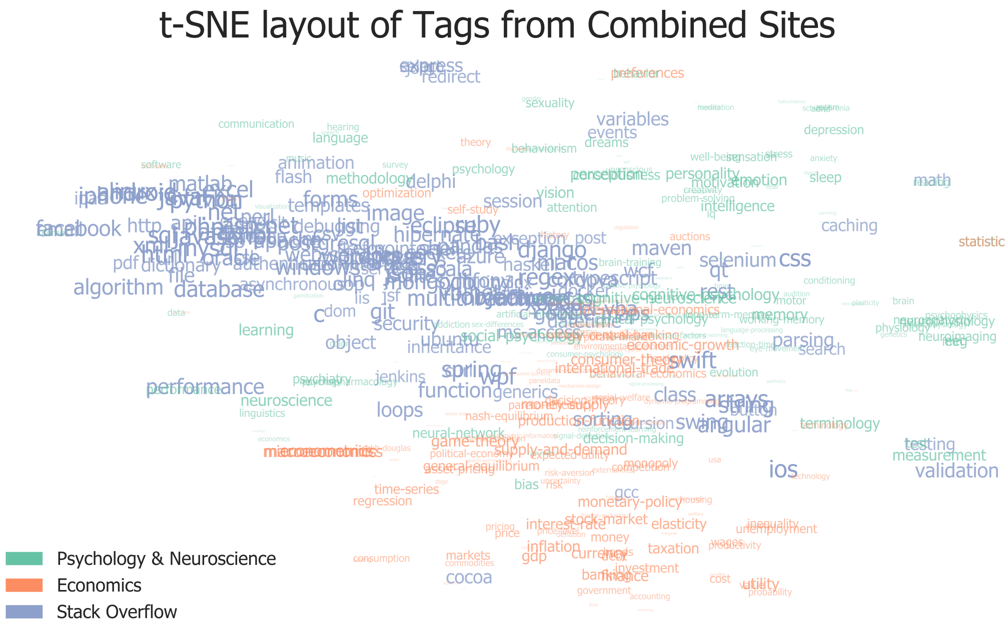


Figure 8: *t-SNE reduction of tags from Stack Overflow, Economics and Psychology & Neuroscience combined, sized by number of usages on the site, notice some tags occur on multiple sites e.g. statistics*

3.3 Comparison

After training, the models from epoch 30 for each subject was selected and ran on all publications in each subject. The resulting percentage of predicted com-

C	Python	Sociology	Statistic	Computational	Network
358–379	c++	anthropology	kolmogorov-smirnov	computation	networks
46:420423	perl	criminology	chi-squared	numerical	network-based
46:849–854	prolog	geography	likelihood-ratio	simulation	system
267–276	java	philosophy	wald	algorithmic	networks-based
53:566–574	c/c++	sociological	chi-square	cpu	networking
545–565	ampl	psychology	two-sample	mathematical	hub
53:387–391	javascript	epistemology	bootstrap	computationally	experiment-tation
:1220–1223	opengl	marxist	t-student	runtime	node
xxx-xxx	matlab	economics	durbin-watson	parallelization	topology
400–416	fortran	historiography	f-test	scalability	graph

Table 5: *Selected words and their 10 nearested neighbours*

putational publications as compared to the number of explicitly computational publications is given in Table 6. We can see that for all but one subject, 'Other Social Sciences', there are predicted to be more computational publications than those just published in computational sources. To verify that the predictions are correct, samples were collected, both randomly and by selecting those at extrema. A proper verification would require multiple coders separately coding hundreds of publications, but in my small sample, the results of the analysis appear satisfactory. Figure 12 has an example of a predicted computational paper from outside the explicitly computational sources, while Figure 13 shows an example of a non-computational paper that the model thinks has signs of being computational. Comparing the two, it is evident that the neural network is not just looking for keywords, since the 'PORTAL-LIBRARIES AND THE ACADEMY' paper has many 'computational' words (see section 3.5 for discussion).

There is also a another model to compare to. The complete model can also produce a table, Table 7. The table for the complete model disagrees with the partial models in many cases, but in *Media and Communication* the difference is significant. The extent to which the disagreement is significant can be explored more thoroughly by application of Cohen's κ (Cohen, 1960). Table 8 gives the κ values for each subject, the results are more pessimistic than the confusion matrices imply, at 0.70 as the highest value and 0.31 the lowest. Applying Landis and Koch's well known heuristic (Landis and Koch, 1977) we can classify the agreements as fair (0.31) to substantial (0.70). Additionally, because Cohen's κ attempts to down weight agreement based on the possibility of random guessing, and as both models are heavily biased negative, the formula is penalizing estimated agreement. Thus, while not perfect, the models can be considerer to have fair agreement, supporting that they are indeed classifying based on the same underlying distribution of word and word vectors.

Another way to compare the models is by sampling and comparing manually. Figures 10 and 11 show two papers where the models disagree.

	Percentage Explicitly Computational	Percentage Predicted Computational	Difference
Psychology	1.50	7.90	6.40
Educational Sciences	10.70	27.50	16.80
Sociology	2.90	7.10	4.20
Political Science	1.90	12.40	10.50
Other Social Sciences	3.40	19.50	16.10
Media and Communication	39.00	36.10	−2.90
Law	0.80	5.70	4.90
Economics and Business	13.20	36.80	23.60

Table 6: *Comparison ratios of explicitly computational publications and those predicted to be computational by individual models*

	Percentage Explicitly Computational	Percentage Predicted Computational	Difference
Psychology	1.46	7.05	5.59
Educational sciences	10.69	33.25	22.56
Sociology	2.90	11.07	8.16
Political science	1.89	12.40	10.51
Other social sciences	3.38	19.43	16.05
Media and communication	38.98	62.60	23.62
Law	0.77	3.55	2.78
Economics and business	13.17	38.59	25.42

Table 7: *Comparison ratios of explicitly computational publications and those predicted to be computational by the full model, note that counts are slightly different due to differences*

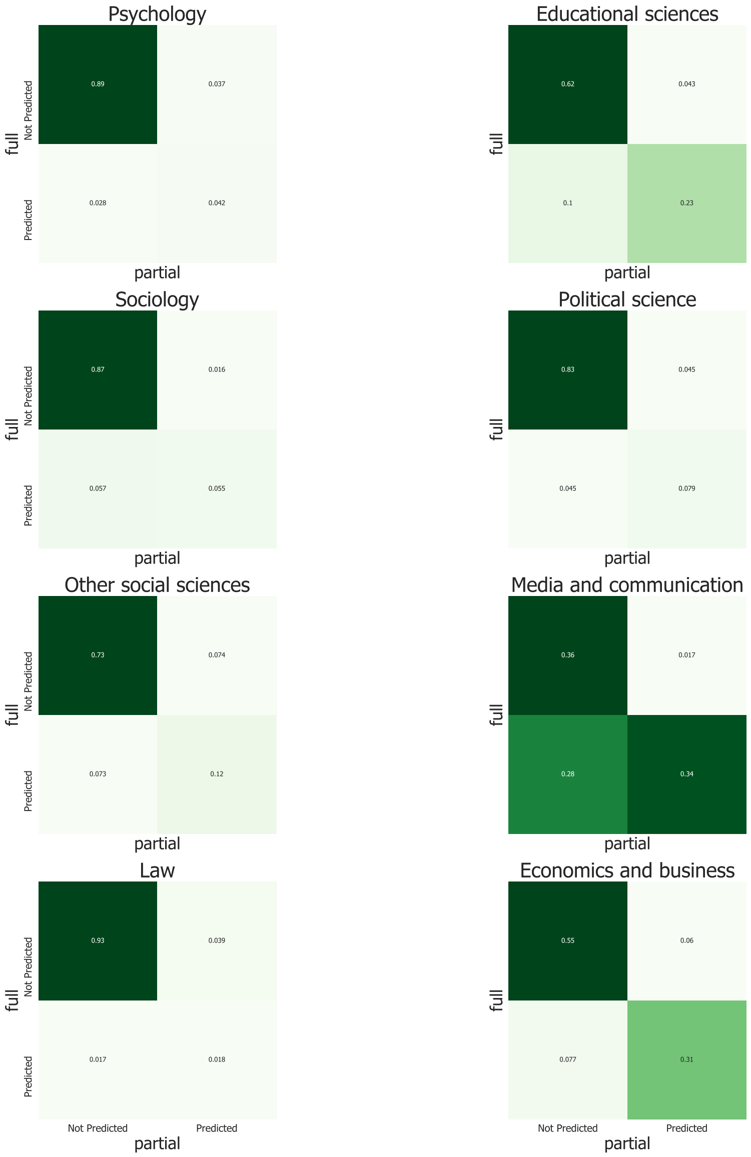


Figure 9: *Confusion matrices of full and subject based classifiers showing the normalized concurrence counts*

Psychology	Educational sciences	Sociology	Political science	Other social sciences	Media and communication	Law	Economics and business
0.50	0.62	0.37	0.31	0.39	0.47	0.31	0.70

Table 8: *Cohen's κ comparing the full and subject based classifiers*

Field	Value
ID	WOS:000206800000005
Explicitly Computational	False
Likelyhood is Comp. Full	45.1%
Likelyhood is Comp. Subject	81.98%
Source	JASSS-THE JOURNAL OF ARTIFICIAL SOCIETIES AND SOCIAL SIMULATION
Year of Publications	2007.0
Title	Higher-Order Simulations: Strategic Investment Under Model-Induced Price Patterns
Abstract	The trading and investment decision processes in financial markets become ever more dependent on the use of valuation and risk models. In the case of risk management for instance, modelling practice has become quite homogeneous and the question arises as to the effect this has on the price formation process. Furthermore, sophisticated

Figure 10: *Example of disagreement between subject and full models*

Field	Value
ID	WOS:000207690200005
Explicitly Computational	False
Likelyhood is Comp. Full	60.4%
Likelyhood is Comp. Subject	12.28%
Source	INTERNATIONAL JOURNAL OF HERITAGE STUDIES
Year of Publications	2008
Title	Place as Dialogue: Understanding and Supporting the Museum Experience
Abstract	This paper presents a dialogical approach to place, people and technology in museums. The approach has been developed in response to concern for locative experience in Interaction Design, an approach to the design and experience of interactive technologies that emphasises the pivotal role played by a wide variety of relationships in experience and suggests a set of dimensions of experience that have been useful in our interpretations of . . .

Figure 11: *Example of disagreement between subject and full models*

3.4 Introspection

Another way to examine if the neural network is working as expected is to examine the output from the RNN section at each word. This is the layers g^2 and h^2 when they are combined. Since the last layer is a shallow one, we know it is working by taking the weighted sum of their outputs so that changes in the outputs lead to similar changes in the final output. Thus, while we cannot say that the value at index 85 is a significant factor, we can say that a change in many indices is correlated, which indicates a change in the output. Figure 14 shows the outputs as the RNN layers read the titles of each of the examples, and a comparison of their final results. By visual inspection, we can see that the word ‘based’ in the positive example had a major impact on the classification while the colon started a major change in the negative example. This suggests that the model has identified ‘based on’ as an indicator of computational papers, which is a valid suggestion. While the model has also identified that ‘: how‘ is an indicator of a negative example. Unfortunately, doing this type of analysis does not have a rigorous backing from the literature, though related work is ongoing (Strobelt et al., 2018). As this type of deep introspection is quite new, there will hopefully be better tools developed in coming years. While this type of visual can be useful, they cannot currently be the basis for decision making, and it remains simply a tool for inspecting the model.

Field	Value
ID	WOS:000318886700008
Explicitly Computational	False
Likelyhood is Computational	81.0%
Source	APPLICATION AND BEST PRACTICE OF COMPETITIVE TECHNICAL INTELLIGENCE
Subject	Media and Communication
Year of Publications	2010
Title	Research on the Key-technology Selection of Virtual Reality Based on Patent Citation Analysis
Citation	Jianmei et al. (2010)
Abstract	Based on the data of ”Derwent Innovation Index” from 1963 to 2009, the authors construct a patent analysis dataset of virtual reality by retrieving the patents through keywords and classification numbers. The authors also reveal technical hot points, key technologies of virtual reality through patent citation analysis and multivariate statistical analysis, and then acquire CTI (competitive technical intelligence) for enterprises . . .

Figure 12: *Example of likely computational publication from a non-computational source*

Field	Value
ID	WOS:000306038900005
Explicitly Computational	False
Likelyhood is Computational	19.1%
Source	PORTAL-LIBRARIES AND THE ACADEMY
Subject	Media and Communication
Year of Publications	2012
Title	Incoming Graduate Students in the Social Sciences: How Much Do They Really Know About Library Research?
Citation	Monroe-Gulick and Petr (2012)
Abstract	Academic librarians provide information literacy instruction and research services to graduate students. To develop evidence-based library instruction and research services for incoming graduate students, the authors interviewed fifteen incoming graduate students in the social sciences and analyzed the interviews using the Association of College & Research Libraries Information Literacy Competency Standards for Higher Education (ACRL Standards). This article discusses the findings, including the authors' assumptions of student information illiteracy, trends noted during the interview analysis, and implications for delivering information literacy training to graduate students in a group discussion modality.

Figure 13: *Example of possible non-computational publication from a non-computational source*

3.5 Word Usage

Figure 15 shows a word cloud constructed from the titles of the computational and non-computational publications in Sociology. The word cloud is constructed by counting the number of occurrences of a word and sizing it according to the count, with the words then laid out algorithmically. This visualization allows us to see what the relationship is between words used in the two sets. We can see in particular that the computational publications refer to 'based' much more than

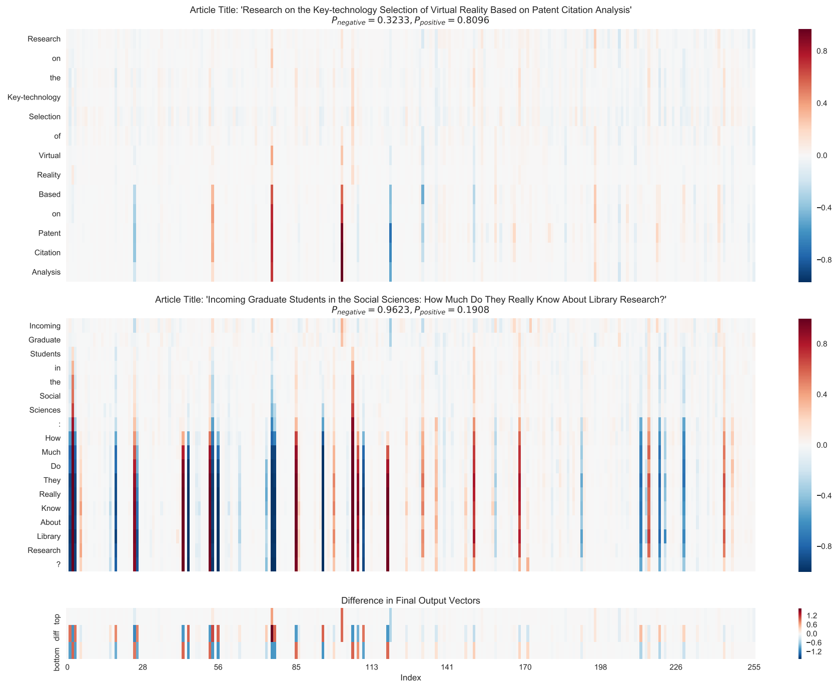


Figure 14: *RNN activations for each word in two titles; the positive example is on top, the negative below it, and a comparison of each input's final output is shown at the bottom*

the non-computational ones, along with other words such as 'Network', 'System' and 'Information', all of which are related to more analytical approaches. While the non-computational publications refer to'Relationship', 'Social' and 'Women', which suggests a more quantitative style of paper. Word clouds for each of the subjects can be found in Appendix B.

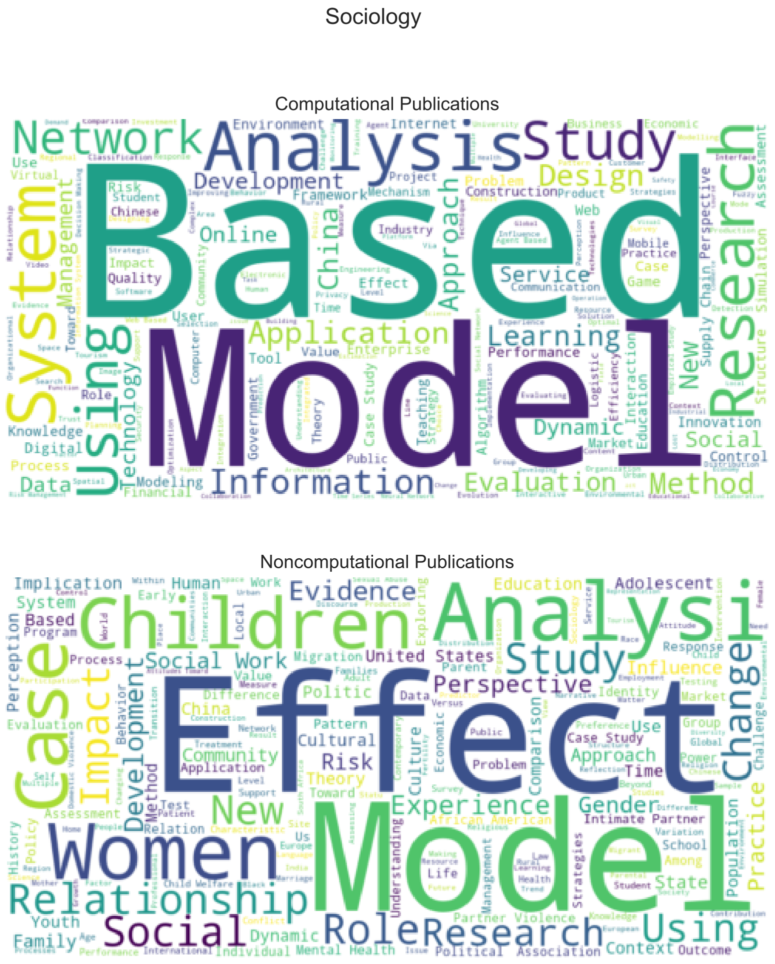


Figure 15: Word cloud of Sociology publication's titles

3.6 Occlusion Effects

A method used to some success in convolutional neural network (CNN) visualization and introspection is the *occlusion experiment*, where an image is broken into subsections and the CNN is asked to make a classification with the subsection removed (Zeiler and Fergus, 2014). This method allows one to measure the significance of each of the subsections by measuring how their removal affects the classification certainty. Figure 16 shows the effects of removing words from the title and abstract on the classification probability for a toy example with the title of *Exploration of humans, society and R* and abstract '*We used methods and techniques to do stuff. Weber Freud and vocabulary acquisition were explored. Then we did more stuff. Our code can be found on Github.*'. We can see in the bottom left that if the title is *Exploration* and the abstract simply *We used methods and techniques to do stuff.* then the full model gives it a 0.64 probability of being computational, but if you add the sentence *Weber Freud and vocabulary acquisition were explored.* the probability drops to 0.41. Similarly adding the final sentence *Our code can be found on Github.* firmly puts the record as computational with a probability of 0.89 and if *R* is added to the title, the probability goes up to 0.95.

This methodology adds support to the RNN classifying based in part on words, but also on word order. If the last sentence is put first in the abstract, the final model probability goes down by more than 20%. This method is explored much further on the interactive webapp hosted at shiny.reidmcy.com/fp. There is one major concern with making inferences with occlusion of this type of model, which is that there is no baseline blank set of words. In image processing, the pixels can be coloured a uniform mixture with 50% brightness (grey), while with word embeddings, there is no such null vector, thus removing the entire vector from the input is the best option. It would be simple to add a 'removed' vector to the training, and that is an intriguing direction of future research.

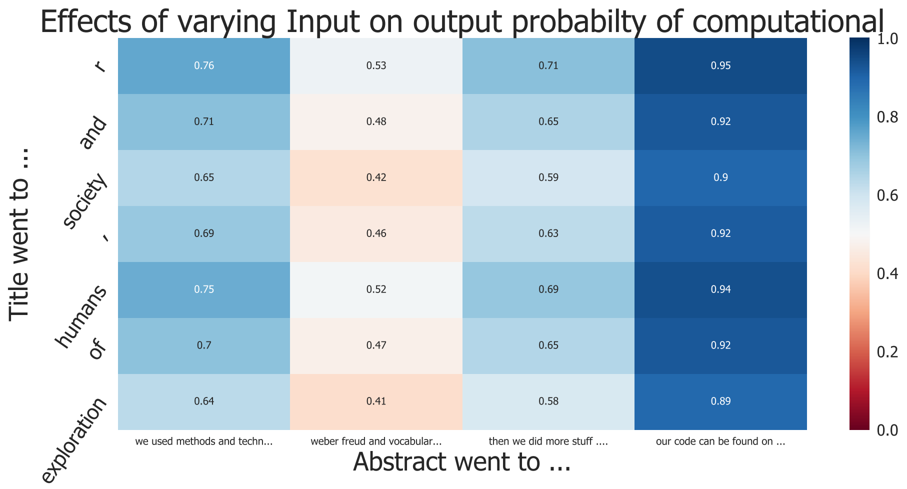


Figure 16: *Model prediction as the title and abstract are added to, colour indicates the probability of the record being computational with blue being high and red low.*

3.7 Temporal Effects

Figure 17 shows the number of computational papers per year for each of the subjects. The peak around 2007 for Media and Communications, and some others, is mostly due to conferences (‘International Conference on Information Management, Innovation Management and Industrial Engineering’ contributes the most) being reclassified from Media and Communications to either a pure computer science or a more general subject (e.g. ‘Multidisciplinary Sciences’) after 2008. What is notable is that the number of computational papers indicated by the model do follow the curve, but with a longer delay. Even as the conferences left the need to publish remained. Additionally, the low level of computational work is present in all disciplines. These are publications like ‘The application of virtual reality technology in interior design system development’ (Chuanrong and Hengliang, 2016), which are discussions of computational techniques by members of the community for the community. In some cases they are like Boyle (Shapin et al., 1985), attempting to layout a new programme of study to a skeptical audience who do not have the capability to observe the phenomena first-hand. In others, the publications are completely integrated into the existing culture and communicating discoveries or designs of machines that are understood by the community (Cetina, 2009). Distinguishing between the two classes of publications is a challenge that will likely be overcome at a later date, but for now I can only say that it seems that social scientists have not changed their discussion of computational techniques by a significant amount in the last decade, baring some decreases in Psychology, Sociology and Economics.

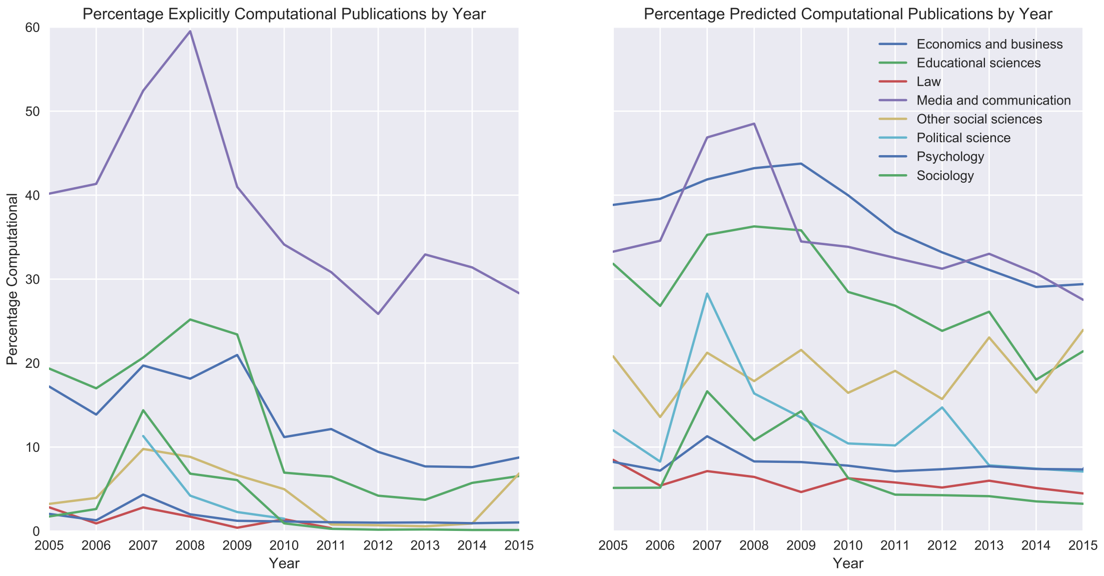


Figure 17: *Yearly percentages of explicitly computational and predicted computational papers, notice the higher base line for the predicted papers and the smoother yearly transitions*

3.8 Authors

Figure 18 shows the smoothed probability density of the number of publications per author across computational and non-computational publications. The number of publications by each author was counted, then their counts were distributed from 0 to 20, with higher counts being cut down to 20 (thus the bump for a few subjects). Then a kernel density estimator with a bandwidth of .5 was used to interpolate and smooth, giving the continuous probability densities displayed.

The difference between the peaks of most computational and non-computational works is not surprising, as most psychologists aren't writing software. What is more surprising is the ones with overlap. High degrees of overlap suggests that computational approaches are an excepted and normal means of doing science for that community. Maybe these computational people are forming their own community with it's own mediums of exchange with the outside world (Star and Griesemer, 1989) or they are integrated into the knowledge society (Cetina, 2009). More research on the subject would be required to answer this.

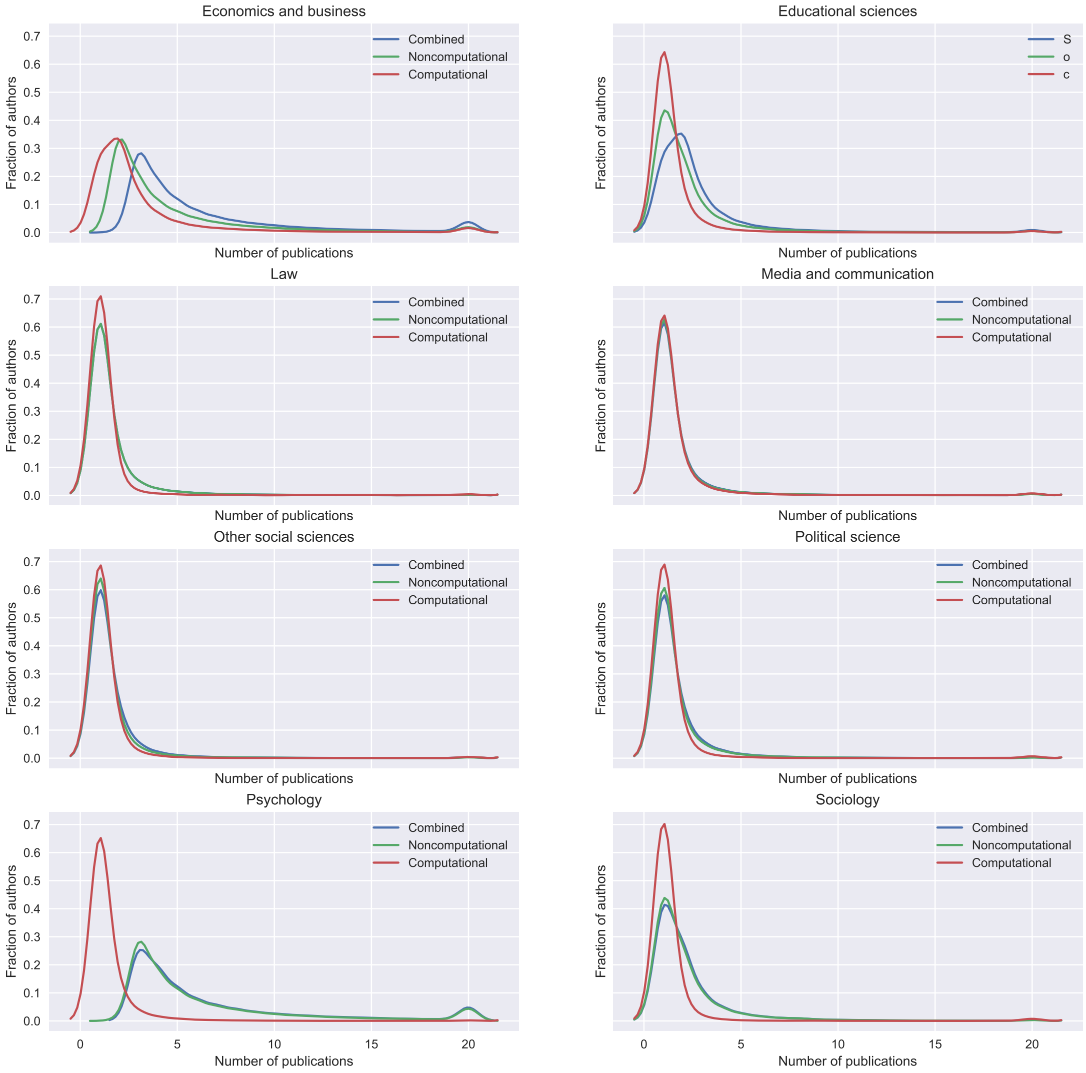


Figure 18: *Probability density of the number of publication for a random author from either the computational, noncomputational or combined pool of publications in a subject*

4 Discussion

4.1 What is being classified

Unlike the more traditional techniques discussed in Section 2.3, deep neural networks' decision making processes are not well understood (Goodfellow et al., 2016a). This limits understanding to inferences based on the observed outputs and inputs. The nominal goal of the classifier describe above is to identify computational usage across disciplines. But that is not what the training and testing data are measuring against. The data provided are collections of works from selected journals. What the journals impose on the papers, in particular upon their abstracts, that is different from the non-computational journals, is what the models are trained to identify. While there are certainly a much higher percentage of computational papers in the journals, as was shown above, they are not the sole holders of the computational keys. It is probably more accurate to characterize the classifications as derived from the style of the abstract and title than the substance.

Then again, what makes a paper computational if not the style. Computer science existed well before computers could be produced. Consider Leibniz's difference engine or Ada Lovelace's Note G. It could be considered that computational social science is as much a style of work as it is a set of methods. In *Bit by Bit* Matthew Salganik (Salganik, 2017) describes many types of computational experiments, but the analysis required for some would not be unfamiliar to Kurt Lewin (Lewin, 1939) as all that changes is that the data are collected without the experimenters leaving their lab. What makes a research topic computational is the style of the approach, collection of all data possible and maybe more complex modeling at the end, not necessarily the substance. Thus looking for computational style might be the correct way to consider the problem of discovering computational usage.

4.2 Results

These results presented are not the definitive word on the distribution of computational methods across the social sciences, but they do represent a strong starting point. That this type of analysis produces anything at all except for noise shows that the deep neural network approach to complex, unstructured text is worth exploring and I plan to continue with it, hopefully improving accuracy and reducing computational complexity.

In addition to the improvements, there are two problems with the current analysis that will need to be addressed. First, the usage of journals as the basis of the initial classification is very coarse, so either a better method needs to be created or more human coding must be done. Unfortunately, for the former method, a simple heuristic or machine learning approach can't be used as then the neural network would learn that instead of the correct distinction (Goodfellow et al., 2016a). Thus, my preferred solution is to reduce the training time and train a series of networks, with the outputs from each generation being used to enrich the next one's training data. This has the downside of requiring a large amount of human intervention though. The other issue with this result is that it cannot be generalized. The usage of word embedding trained on the complete universe of interest means that if a new record is to be considered, the embedding has to be redone, or at least updated, which in either case requires a complete retraining of the model. Even with these two issues, there are some useful results, and both can be mediated with further work.

4.3 Further Steps

The baseline activity in computation across all subjects is very intriguing, as it suggests that computational approaches may not be as divisive as some suggest (Watts, 2007)(Lazer et al., 2009). There is already precedent for scientists to accept and use new tools, machinery or ideas, without having to change their paradigm. The Latourian idea of a black box (Latour, 1987) can be used to describe many computational techniques. Do everyday scientists care about how the line of the linear regression is derived? No. They only care about the fact that it is accurate, and so why would it matter if a graduate student or a computer made it? The techniques may be more complicated than a linear regression, but that may require a new 'sex' added to the seven sexes of Collins (Collins, 1975), focused on complex computational systems required for replication. Will it also require a fundamental shift in methodology?

An additional point of particular interest is 'how do new styles/methods diffuse through science?' There is much work already on the diffusion of knowledge (Griliches, 1960) (Crane, 1972) (Evans, 2010), but there is much less work on diffusion of style or technique. The data provided by this method combined with bibliographic analysis would likely reveal some interesting results. Additionally, the method is not limited to identifying computational approaches it can be trivially generalized given the correct training data. Of note, instead of binary classification, a mixture model could be created, where different styles exist in different proportions in each record thus along the competition of styles to be considered.

There are a few more aspects, particular to Science Studies, that were not able to be a part of this analysis. First, examining the impact of gender on computational publications. Women's representation in software development is low, but is it also low in scientific computation? Secondly, how do computational researchers fit into the modes of science already in place, e.g. can computational sociology coexist with the strong programme (Bloor, 1976)? This work suggests that they can fit in well. Finally, is the advent of computational social science a paradigm shift? Or is just another set of black box to help scientists to get slightly closer to the truth?

References

Anderson, Chris. 2008. "The end of theory: The data deluge makes the scientific method obsolete." *Wired magazine* 16:16–07.

Back, Mitja D, Albrecht CP Küfner, and Boris Egloff. 2010. "The emotional timeline of September 11, 2001." *Psychological Science* 21:1417–1419.

Berners-Lee, Tim, Robert Cailliau, Jean-François Groff, and Bernd Pollermann. 2010. "World-wide web: The information universe." *Internet Research* 20:461–471.

Bird, Steven. 2006. "NLTK: the natural language toolkit." In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pp. 69–72. Association for Computational Linguistics.

Bloor, David. 1976. "The strong programme in the sociology of knowledge." *Knowledge and social imagery* 2:3–23.

Bolukbasi, Tolga, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. "Man is to computer programmer as woman is to homemaker? Debiasing word embeddings." In *Advances in Neural Information Processing Systems*, pp. 4349–4357.

Börner, Katy. 2010. *Atlas of Science: Visualizing What We Know*. Cambridge: MIT Press.

Börner, Katy. 2015. *Atlas of Knowledge: Anyone Can Map*. Cambridge: MIT Press.

Boyack, Kevin, Richard Klavans, and Katy Börner. 2005. "Mapping the Backbone of Science." *Scientometrics* 64:351–374.

Cetina, Karin Knorr. 2009. *Epistemic cultures: How the sciences make knowledge*. Harvard University Press.

Chetlur, Sharan, Cliff Woolley, Philippe Vandermersch, Jonathan Cohen, John Tran, Bryan Catanzaro, and Evan Shelhamer. 2014. "cudnn: Efficient primitives for deep learning." *arXiv preprint arXiv:1410.0759* .

Chu, Johan SG and James A Evans. 2018. "Too Many Papers? Slowed Canonical Progress in Large Fields of Science." .

Chuanrong, Chen and Tang Hengliang. 2016. “The Application of Digital Technology in Interior Design Education.” In *Measuring Technology and Mechatronics Automation (ICMTMA), 2016 Eighth International Conference on*, pp. 688–691. IEEE.

Clarivate Analytics. 2016. “Acquisition of the Thomson Reuters Intellectual Property and Science Business by Onex and Baring Asia Completed.”

Cohen, Jacob. 1960. “A coefficient of agreement for nominal scales.” *Educational and psychological measurement* 20:37–46.

Collins, Harry M. 1975. “The seven sexes: A study in the sociology of a phenomenon, or the replication of experiments in physics.” *Sociology* 9:205–224.

Collins, Robert J, Paul Joseph Apodaca, Adam J Wand, and Claude Jones III. 2012. “Advertiser reporting system and method in a networked database search system.” US Patent 8,321,275.

Crane, Diana. 1972. “Invisible colleges; diffusion of knowledge in scientific communities.” .

De Bellis, Nicola. 2009. *Bibliometrics and citation analysis: from the science citation index to cybermetrics*. Scarecrow Press.

De Jong, Hidde and Arie Rip. 1997. “The computer revolution in science: Steps towards the realization of computer-supported discovery environments.” *Artificial intelligence* 91:225–256.

Efremenkova, VM and SM Gonnova. 2016. “A comparison of Scopus and WoS database subject classifiers in mathematical disciplines.” *Scientific and Technical Information Processing* 43:115–122.

Evans, James and Jacob Foster. 2011. “Metaknowledge.” *Science* 331:721–725.

Evans, James A. 2010. “Industry collaboration, scientific sharing, and the dissemination of knowledge.” *Social Studies of Science* 40:757–791.

Evans, James A and Pedro Aceves. 2016. “Machine translation: mining text for social theory.” *Annual Review of Sociology* 42:21–50.

Foster, Jacob G, Andrey Rzhetsky, and James A Evans. 2015. “Tradition and innovation in scientists’ research strategies.” *American Sociological Review* 80:875–908.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. 2016a. *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. 2016b. *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Gordon, Robert J. 2000. “Does the” new economy” measure up to the great inventions of the past?” Technical report, National bureau of economic research.

Graves, Alex, Navdeep Jaitly, and Abdel-rahman Mohamed. 2013. “Hybrid speech recognition with deep bidirectional LSTM.” In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pp. 273–278. IEEE.

Graves, Alex and Jürgen Schmidhuber. 2005. “Framewise phoneme classification with bidirectional LSTM and other neural network architectures.” *Neural Networks* 18:602–610.

Greff, Klaus, Rupesh K Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. 2017. “LSTM: A search space odyssey.” *IEEE transactions on neural networks and learning systems* 28:2222–2232.

Griliches, Zvi. 1960. “Hybrid corn and the economics of innovation.” *Science* 132:275–280.

Hochreiter, Sepp and Jürgen Schmidhuber. 1997. “Long Short-Term Memory.” *Neural Comput.* 9:1735–1780.

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2013. *An introduction to statistical learning*, volume 112. Springer.

Jianmei, Wang, Jin Xuehui, and Meng Hong. 2010. “Research on the Key-technology Selection of Virtual Reality Based on Patent Citation Analysis.”

John Walker, Saint. 2014. “Big data: A revolution that will transform how we live, work, and think.”

Jurafsky, Dan and James H Martin. 2014. *Speech and language processing*, volume 3. Pearson London.

Kiss, Tibor and Jan Strunk. 2006. “Unsupervised multilingual sentence boundary detection.” *Computational Linguistics* 32:485–525.

Knowledge Lab. 2017. “Web Of Science (WoS) — Cloud Kotta beta documentation.”

Kossinets, Gueorgi and Duncan J Watts. 2006. "Empirical analysis of an evolving social network." *science* 311:88–90.

Kramer, Adam DI, Jamie E Guillory, and Jeffrey T Hancock. 2014. "Experimental evidence of massive-scale emotional contagion through social networks." *Proceedings of the National Academy of Sciences* 111:8788–8790.

Landauer, Thomas K. 1995. *The trouble with computers: Usefulness, usability, and productivity*, volume 21. Taylor & Francis.

Landis, J Richard and Gary G Koch. 1977. "The measurement of observer agreement for categorical data." *biometrics* pp. 159–174.

Latour, Bruno. 1987. *Science in action: How to follow scientists and engineers through society*. Harvard university press.

Lazer, David, Ryan Kennedy, Gary King, and Alessandro Vespignani. 2014. "The parable of Google Flu: traps in big data analysis." *Science* 343:1203–1205.

Lazer, David, Alex Sandy Pentland, Lada Adamic, Sinan Aral, Albert Laszlo Barabasi, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, et al. 2009. "Life in the network: the coming age of computational social science." *Science (New York, NY)* 323:721.

Levy, Omer and Yoav Goldberg. 2014. "Neural word embedding as implicit matrix factorization." In *Advances in neural information processing systems*, pp. 2177–2185.

Lewin, Kurt. 1939. "Field theory and experiment in social psychology: Concepts and methods." *American journal of sociology* 44:868–896.

Maaten, Laurens van der and Geoffrey Hinton. 2008. "Visualizing data using t-SNE." *Journal of machine learning research* 9:2579–2605.

Marcus, Mitchell P, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. "Building a large annotated corpus of English: The Penn Treebank." *Computational linguistics* 19:313–330.

McCallum, Andrew, Kamal Nigam, et al. 1998. "A comparison of event models for naive bayes text classification." In *AAAI-98 workshop on learning for text categorization*, volume 752, pp. 41–48. Citeseer.

McLevey, John and Reid McIlroy-Young. 2016. "metaknowledge documentation." `http://networkslab.org/metaknowledge/documentation/metaknowledgeFull.html#WOSRecord`.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. "Efficient estimation of word representations in vector space." *arXiv preprint arXiv:1301.3781* .

Monroe-Gulick, Amalia and Julie Petr. 2012. "Incoming graduate students in the social sciences: how much do they really know about library research?" *portal: Libraries and the Academy* 12:315–335.

Padgett, John F and Christopher K Ansell. 1993. "Robust Action and the Rise of the Medici, 1400-1434." *American journal of sociology* 98:1259–1319.

Pfaffenberger, Bryan. 1988. "The social meaning of the personal computer: Or, why the personal computer revolution was no revolution." *Anthropological Quarterly* pp. 39–47.

Provost, Foster and Tom Fawcett. 2013. "Data science and its relationship to big data and data-driven decision making." *Big data* 1:51–59.

PyTorch core team. 2017. *PyTorch*.

Řehůřek, Radim and Petr Sojka. 2010. "Software Framework for Topic Modelling with Large Corpora." In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pp. 45–50, Valletta, Malta. ELRA. `http://is.muni.cz/publication/884893/en`.

Riedmiller, Martin and Heinrich Braun. 1993. "A direct adaptive method for faster backpropagation learning: The RPROP algorithm." In *Neural Networks, 1993., IEEE International Conference on*, pp. 586–591. IEEE.

Rodriguez, Paul, Janet Wiles, and Jeffrey L Elman. 1999. "A recurrent neural network that learns to count." *Connection Science* 11:5–40.

Salganik, Matthew J. 2017. *Bit by bit: social research in the digital age*. Princeton University Press.

Shapin, Steven, Simon Schaffer, and Thomas Hobbes. 1985. *Leviathan and the air-pump*. Princeton University Press Princeton.

Shi, Feng, Jacob Foster, and James Evans. 2015. "Weaving the fabric of science: Dynamic network models of science's unfolding structure." *Social Networks* 43:73–85.

Skupin, André, Joseph Biberstine, and Katy Börner. 2013. "Visualizing the topical structure of the medical sciences: a self-organizing map approach." *PloS one* 8:e58779.

Stack Exchange network. 2018. *Stack Exchange Data Dump*.

Star, Susan Leigh and James R Griesemer. 1989. "Institutional ecology,translations' and boundary objects: Amateurs and professionals in Berkeley's Museum of Vertebrate Zoology, 1907-39." *Social studies of science* 19:387–420.

Strobelt, Hendrik, Sebastian Gehrmann, Hanspeter Pfister, and Alexander M Rush. 2018. "Lstmvis: A tool for visual analysis of hidden state dynamics in recurrent neural networks." *IEEE transactions on visualization and computer graphics* 24:667–676.

Sugimoto, Cassidy, Vincent Lariviere, Chaoqun Ni, Yves Gingras, and Blaise Cronin. 2013. "Global gender disparities in science." *Nature* 504:211–213.

Thomson Reuters. 2012. "OECD CATEGORY SCHEME." `http://ipscience-help.thomsonreuters.com/incitesLive/globalComparisonsGroup/globalComparisons/subjAreaSchemesGroup/oecd.html`.

Watts, Duncan J. 2007. "A twenty-first century science." *Nature* 445:489–489.

Weizenbaum, Joseph. 1972. "On the Impact of the Computer on Society." *Science* 176:609–614.

Werbos, Paul J. 1982. "Applications of advances in nonlinear sensitivity analysis." In *System modeling and optimization*, pp. 762–770. Springer.

Zeiler, Matthew D and Rob Fergus. 2014. "Visualizing and understanding convolutional networks." In *European conference on computer vision*, pp. 818–833. Springer.

A Appendix: Complete table of WOS Social Science subject tags

Subject	WOS Code	Subsubject
5.01 Psychology	BV	PSYCHOLOGY, BIOLOGICAL
5.01 Psychology	CN	BEHAVIORAL SCIENCES
5.01 Psychology	HI	PSYCHOLOGY, EDUCATIONAL
5.01 Psychology	JI	ERGONOMICS
5.01 Psychology	MY	PSYCHOLOGY, DEVELOPMENTAL
5.01 Psychology	NQ	PSYCHOLOGY, APPLIED
5.01 Psychology	VI	PSYCHOLOGY
5.01 Psychology	VJ	PSYCHOLOGY, MULTIDISCIPLINARY
5.01 Psychology	VS	PSYCHOLOGY, MATHEMATICAL
5.01 Psychology	VX	PSYCHOLOGY, EXPERIMENTAL
5.01 Psychology	WQ	PSYCHOLOGY, SOCIAL
5.02 Economics and business	DI	BUSINESS
5.02 Economics and business	DK	BUSINESS, FINANCE
5.02 Economics and business	GY	ECONOMICS
5.02 Economics and business	NM	INDUSTRIAL RELATIONS & LABOR
5.02 Economics and business	PC	MANAGEMENT
5.02 Economics and business	PE	OPERATIONS RESEARCH & MANAGEMENT SCIENCE
5.03 Educational sciences	HA	EDUCATION & EDUCATIONAL RESEARCH
5.03 Educational sciences	HB	EDUCATION, SCIENTIFIC DISCIPLINES
5.03 Educational sciences	HE	EDUCATION, SPECIAL
5.04 Sociology	BF	ANTHROPOLOGY
5.04 Sociology	FU	DEMOGRAPHY
5.04 Sociology	JM	ETHNIC STUDIES
5.04 Sociology	JO	FAMILY STUDIES
5.04 Sociology	PS	SOCIAL SCIENCES, MATHEMATICAL METHODS
5.04 Sociology	WM	SOCIAL ISSUES
5.04 Sociology	WY	SOCIAL WORK
5.04 Sociology	XA	SOCIOLOGY
5.04 Sociology	ZK	WOMEN'S STUDIES
5.05 Law	FE	CRIMINOLOGY & PENOLOGY
5.05 Law	OM	LAW
5.06 Political science	OE	INTERNATIONAL RELATIONS
5.06 Political science	UU	POLITICAL SCIENCE
5.06 Political science	VM	PUBLIC ADMINISTRATION
5.07 Social and economic geography	BM	AREA STUDIES
5.07 Social and economic geography	JB	ENVIRONMENTAL STUDIES
5.07 Social and economic geography	KU	GEOGRAPHY
5.07 Social and economic geography	UQ	PLANNING & DEVELOPMENT
5.07 Social and economic geography	YQ	TRANSPORTATION
5.07 Social and economic geography	YY	URBAN STUDIES
5.08 Media and communication	EU	COMMUNICATION
5.08 Media and communication	NU	INFORMATION SCIENCE & LIBRARY SCIENCE
5.09 Other social sciences	MW	HOSPITALITY, LEISURE, SPORT & TOURISM
5.09 Other social sciences	OR	ASIAN STUDIES
5.09 Other social sciences	EN	CULTURAL STUDIES
5.09 Other social sciences	WU	SOCIAL SCIENCES, INTERDISCIPLINARY

Table 9: *Complete table of WOS Social Science subject tags*

B Appendix: Word Clouds

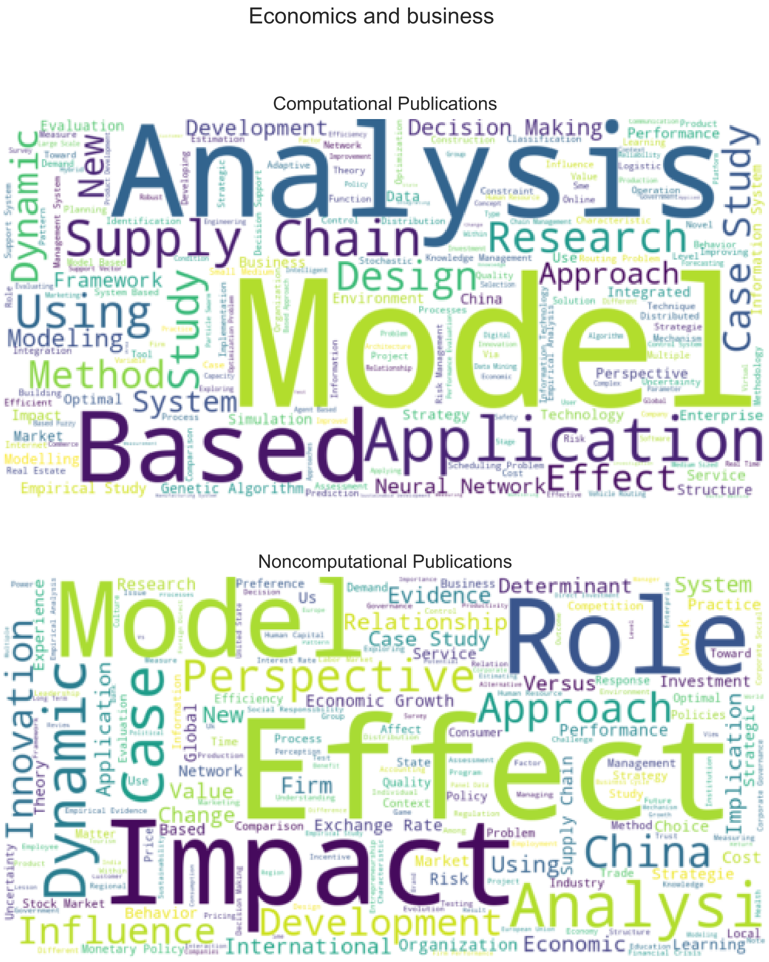
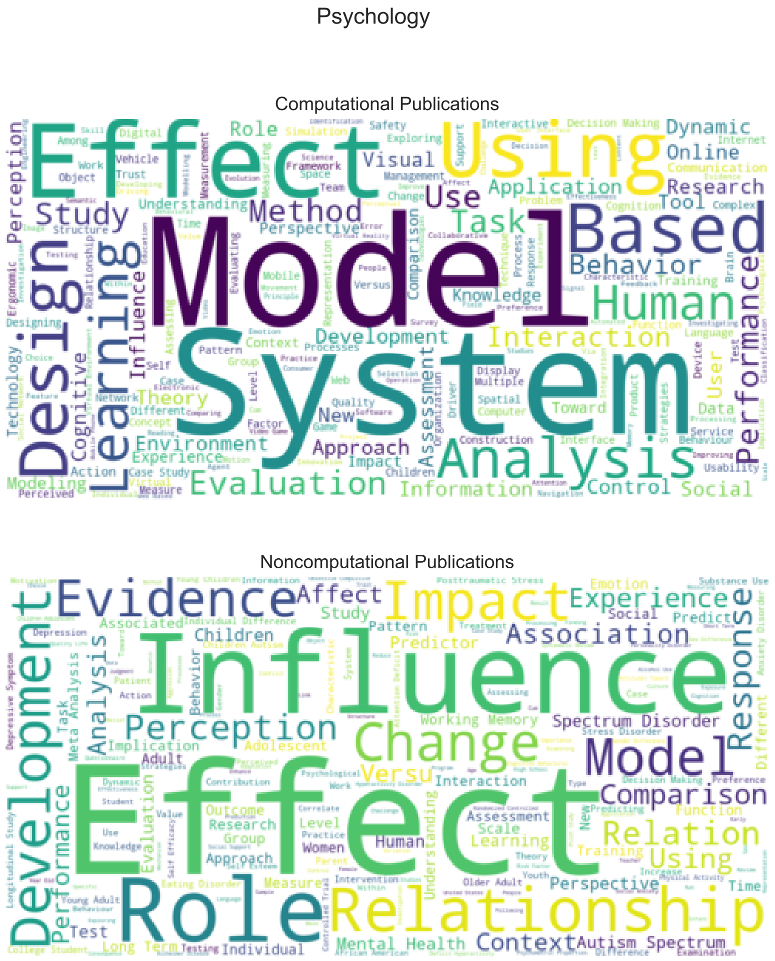


Figure 19: *Word cloud of Economics And Business publication's titles*

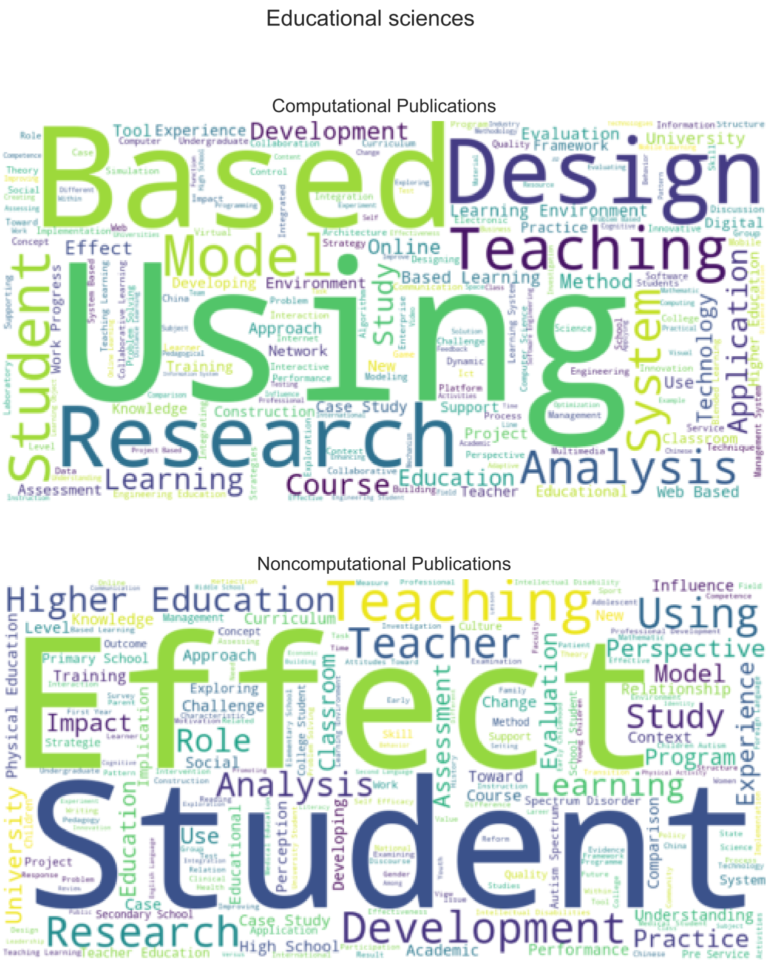


Psychology

Computational Publications

Noncomputational Publications

Figure 20: *Word cloud of Psychology publication's titles*

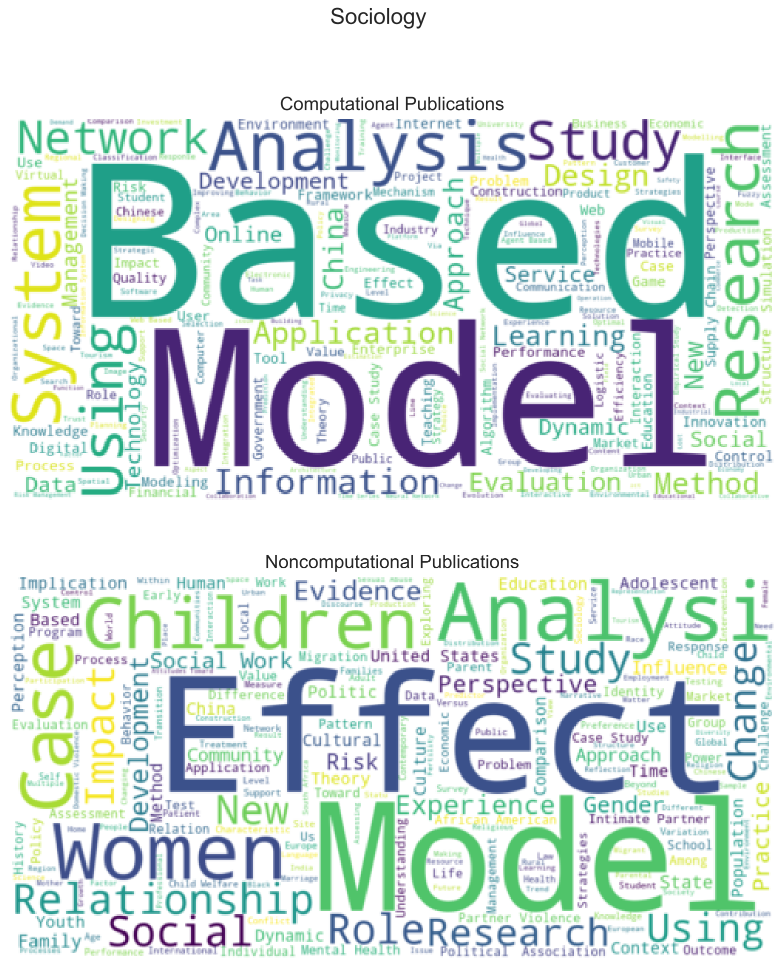


Educational sciences

Computational Publications

Noncomputational Publications

Figure 21: *Word cloud of Educational Sciences publication's titles*

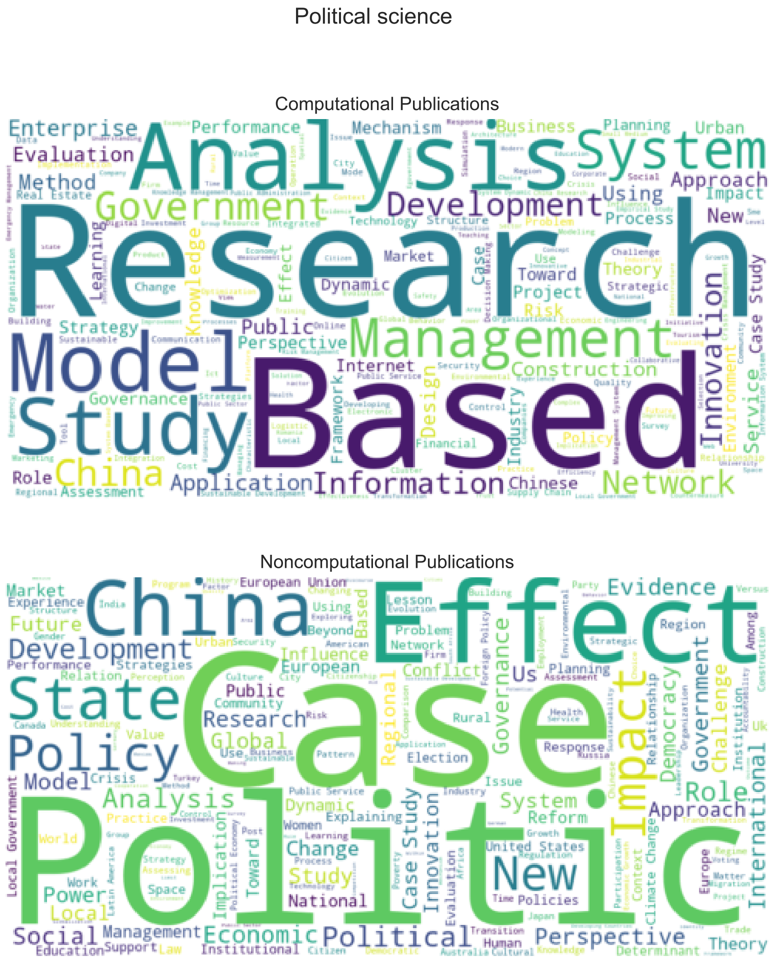


Sociology

Computational Publications

Noncomputational Publications

Figure 22: *Word cloud of Sociology publication's titles*

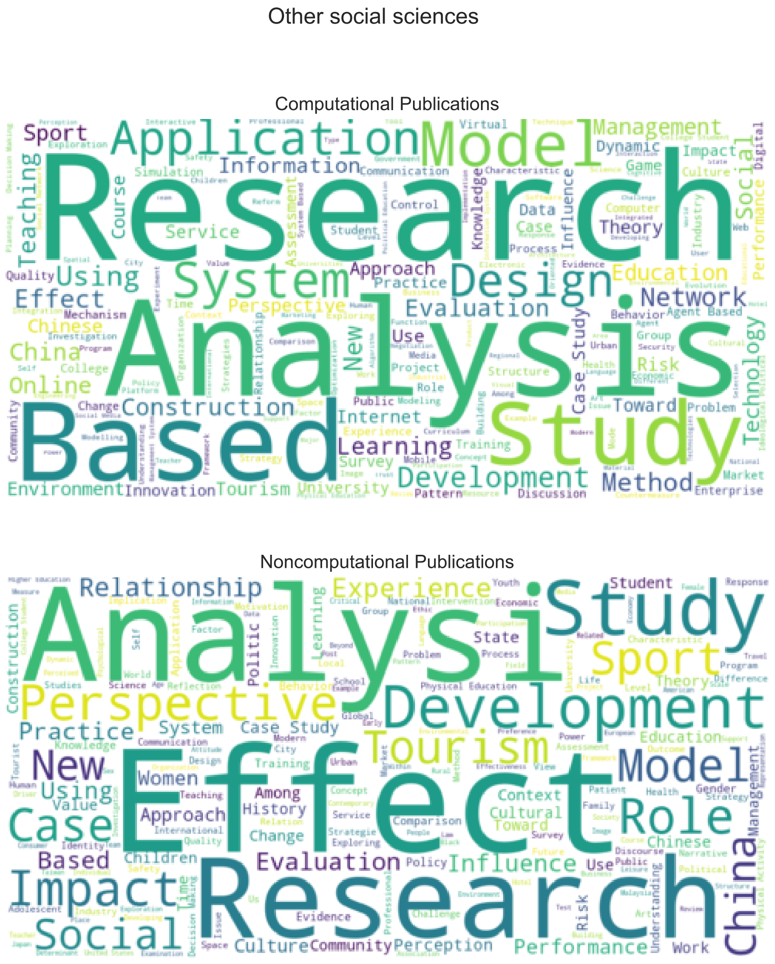


Political science

Computational Publications

Noncomputational Publications

Figure 23: *Word cloud of Political Science publication's titles*



Other social sciences

Computational Publications

Noncomputational Publications

Figure 24: *Word cloud of Other Social Sciences publication's titles*

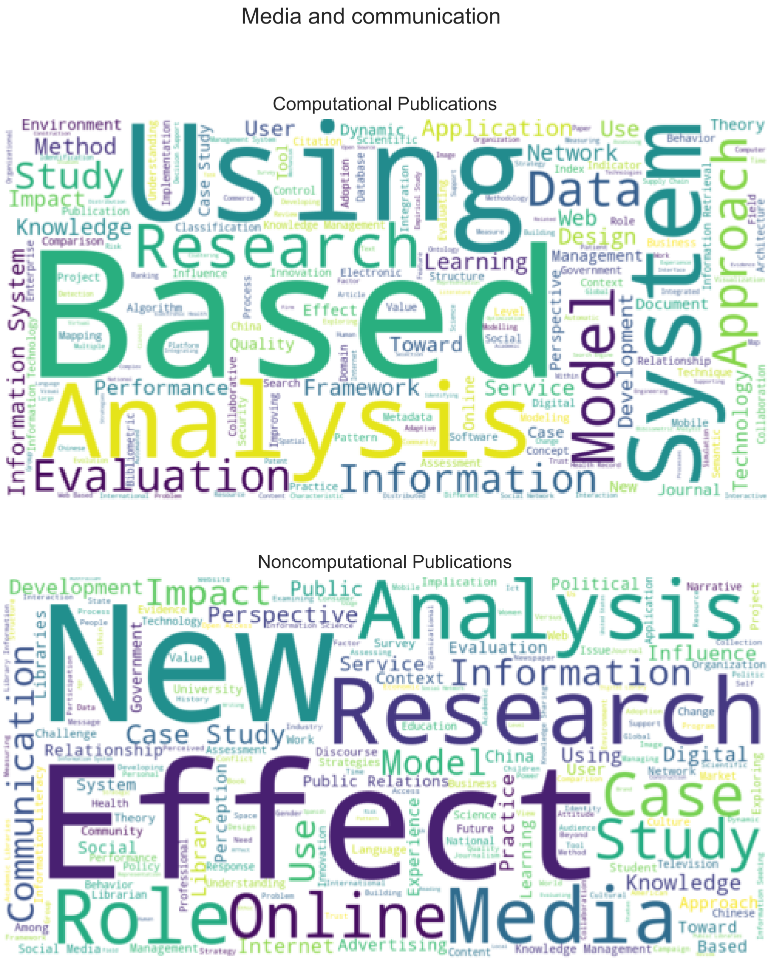
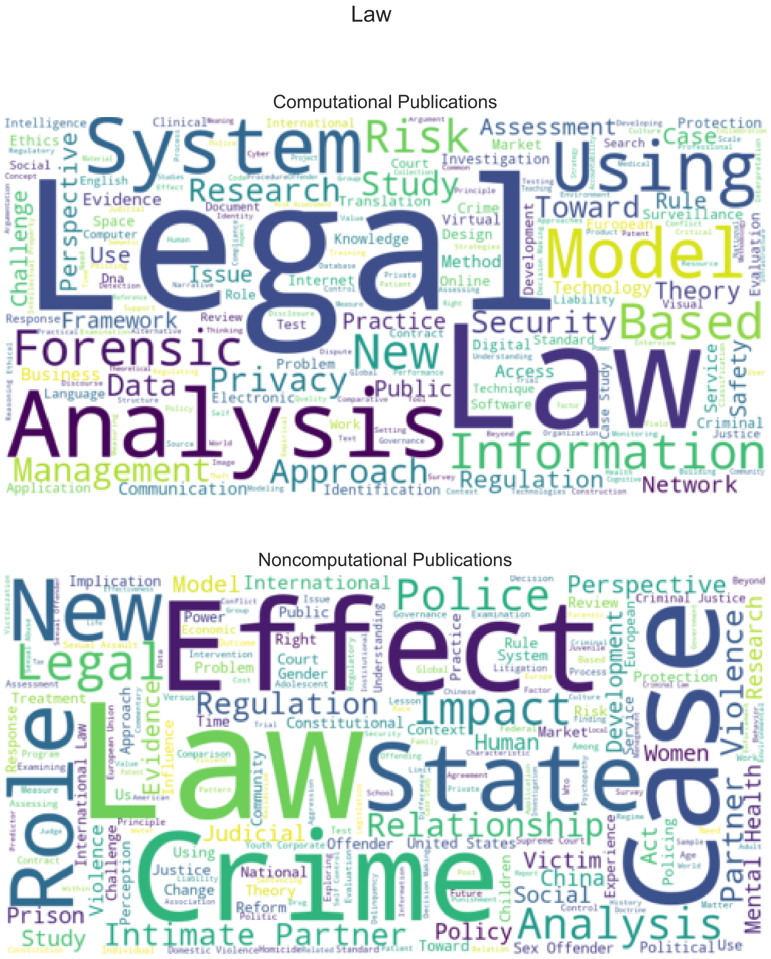


Figure 25: *Word cloud of Media And Communication publication's titles*



Law

Computational Publications

Noncomputational Publications

Figure 26: *Word cloud of Law publication's titles*

C Appendix: Word2Vec t-SNE

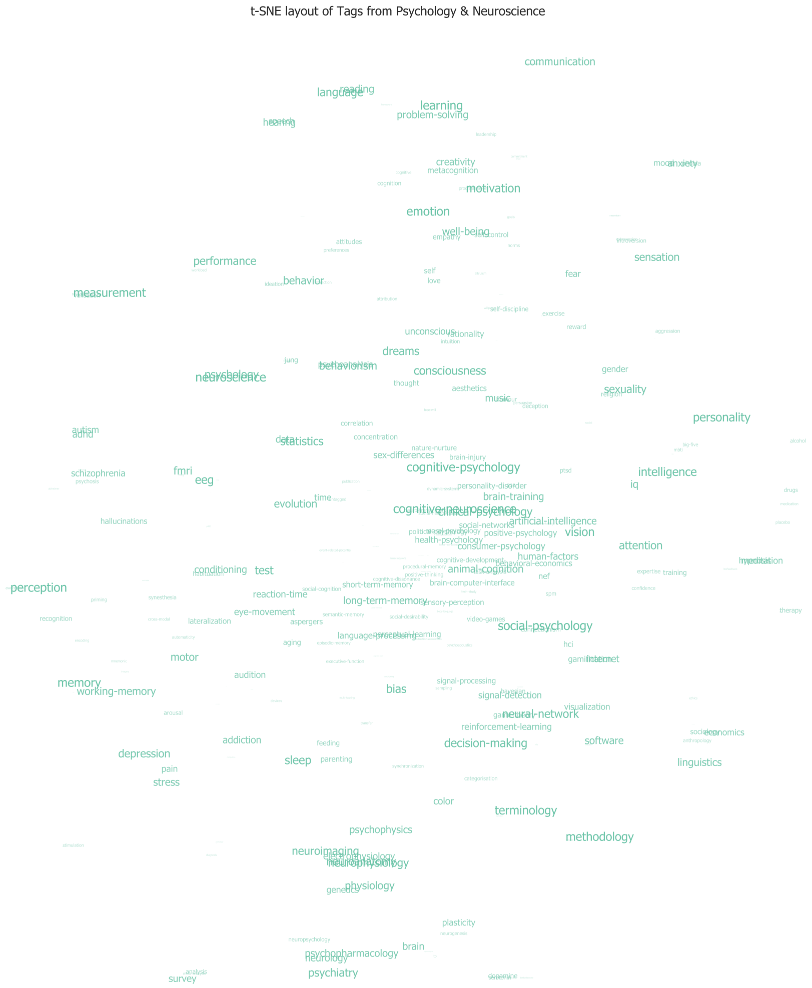


Figure 27: *t-SNE layout of Tags from Psychology & Neuroscience*

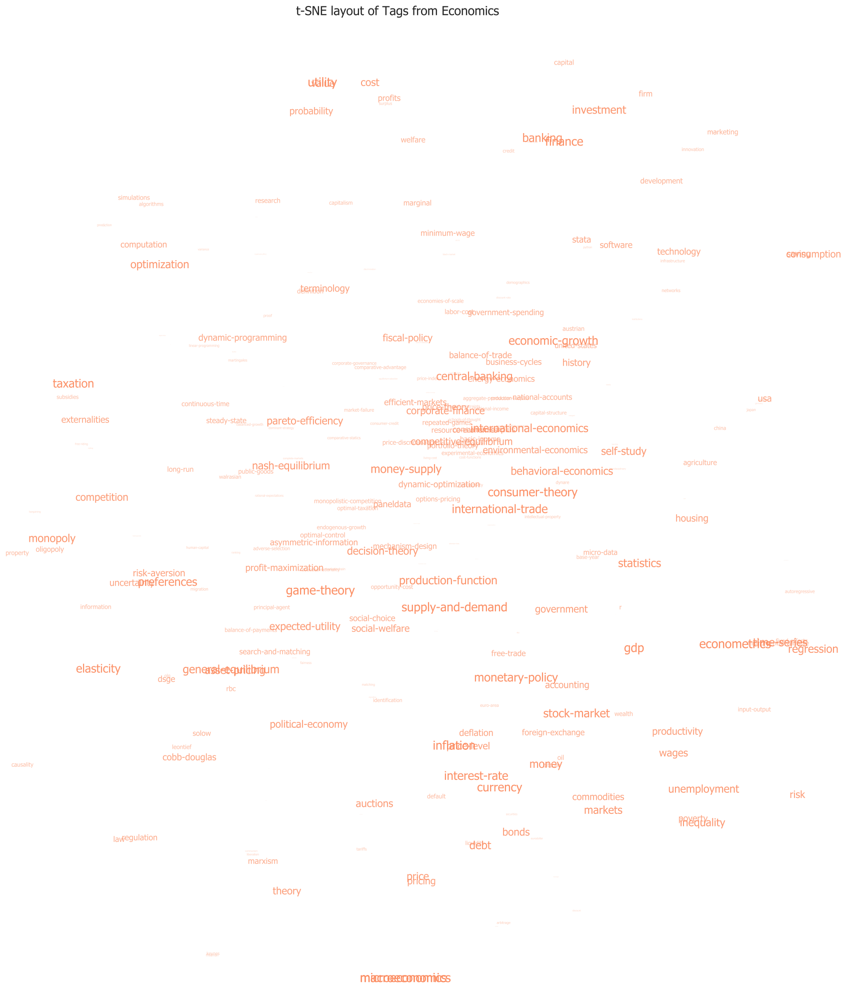


Figure 28: *t-SNE layout of Tags from Economics*

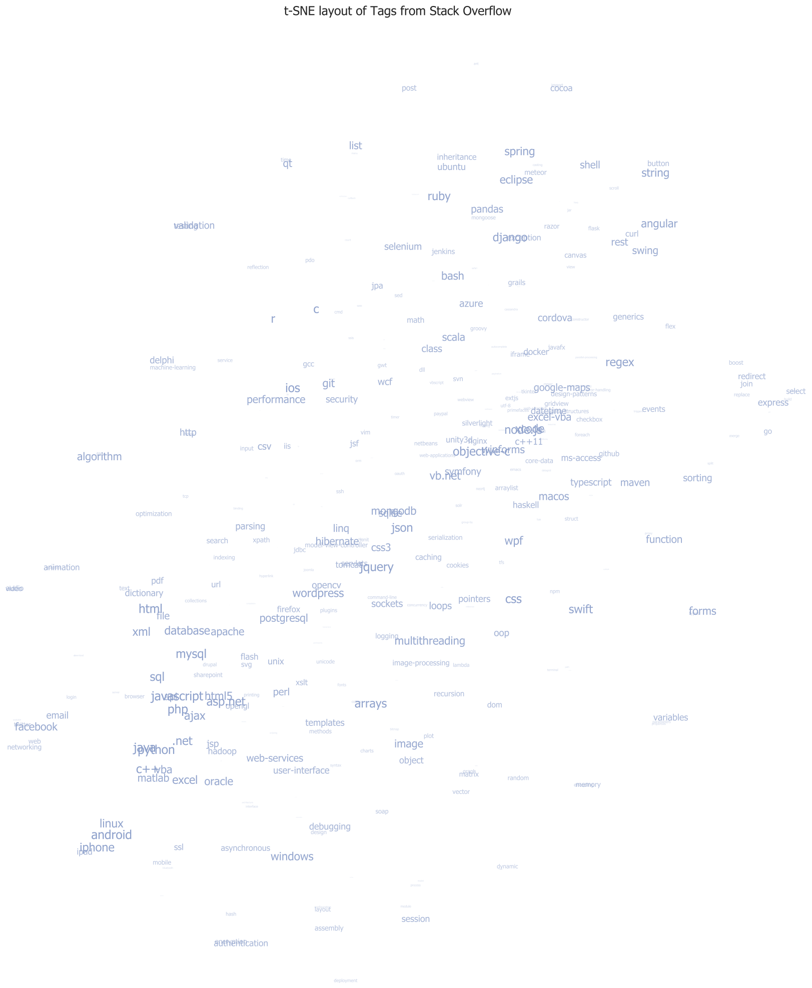


Figure 29: *t-SNE layout of Tags from Stack Overflow*