Universität Koblenz - Landau
Department für Informatik,
Institute for Web Science and Technologies
Prof. Dr. Steffen Staab
Dr. Chandan Kumar
Lukas Schmelzeisen

10. Juli 2018

Master Thesis Kandhasamy Rajasekaran

Fake news classification through Wikipedia using recurrent neural networks

Abstract

The unprecedented growth in production and dissemination of information leads to an unprecedented growth in production and dissemination of fake news. Fake news hinders the society from progress by delaying the pursuit of right information. It is very essential to have a mechanism to detect and control fake news. Several organizations uses collaborative efforts of domain experts, a manual process which cannot withstand the proliferation of news production and dissemination. This research work will use Wikipedia as a ground reality and cross check claims automatically. Recurrent neural networks will be used to understand Wikipedia and the performance of different configurations of Neural Networks will be benchmarked against each other and the already available automated fake news detectors.

1 Introduction

According to Lazer et al.[LBB+18], Fake news is a false information, constructed intentionally (disinformation) or unintentionally (misinformation) and the publishers does not have rigorous news media's editorial norms for making sure of accuracy and credibility. Fake news is prevalent and a research study from Allcot and Gentzkow [AG17] state that "average American adult saw on the order of one or perhaps several fake news stories in the months around the election, with just over half of those who recalled seeing them believing them". A large-scale empirical study with twitter dataset by Vosoughi et al. [VRA18], reveals that fake news spread longer, faster, deeper and broader than the legitimate news. It also reveals that the effects of fake news are more prominent in a political context than news about terrorism, natural disaster, science and other domains.

Most often fake news is detected by people and organizations through their common sense. There are many facts checking websites such as for e.g. snopes.com, factcheck.org, politifact.com etc. which uses collaborative effort of domain experts. In the websites, each news is tagged with a fact meter to refer their authenticity. The main strategy to handle fake news is to check against a reliable source of information and claim its integrity. Although this is fairly good, since it requires manual effort, it is not available for all domains and not scalable. Also, it is unmatchable to the rate at which the information is produced because of many social networks, blogging sites etc.

Wikipedia is a free online encyclopedia available in more than 300 languages with a principle that anyone can edit [Wal05]. It has got a wide range of domains covered with many articles written under each domain. According to Alexa[1] and SimilarWeb[2], Wikipedia is considered to be the fifth

[1] https://www.alexa.com
[2] https://www.similarweb.com

most popular website. According to Wikipedia, the English Wikipedia consists the highest number of articles at present amounting to approximately 500 million. The range of subject covered is wide such as for e.g. Art, Culture, Science, Mathematics, Religion etc. and it is maintained by many authors. The frequency of the update in English Wikipedia is very high and it is approximately equal to 10 updates per second and 600 articles per day. Although it can be edited by anyone, an investigation carried out by Nature[3], reveals that the quality of content is similar to another encyclopedia such as Britannica[4] [Wal05]. Although there can be malicious users in Wikipedia, the culture and the community ensures most of the high impactful errors are rectified very quickly[PCL+07]. Thus English Wikipedia can be used as a proxy for a reliable source of information since most of the content is true.

Understanding Wikipedia is a complex task since it requires understanding the subtleties of English natural language and the context. Machine Learning is the capability of systems learning patterns from raw data. Different features of raw data need to be extracted separately and fed to Machine learning algorithms to get good performance. This drawback is solved by using Deep Learning, which uses neural networks, a multi-layer network of simple representations to learn complex data representations and then extracts patterns out of the data[GBC16]. Deep Learning provides state-of-the-art results in the field such as image recognition, speech recognition, and natural language processing.

The focus of the master thesis is to use Wikipedia as a ground reality or as a source of experts opinion and use this knowledge to cross-check claims automatically. Whether the information is present in Wikipedia or not will be used as a proxy for information being considered as true or fake. Recurrent neural networks with different configurations will be used to understand Wikipedia and the results of each will be compared against each other to understand better.

2 Related work

Many efforts in research have been put in to detect fake news in microblogging platforms such as Twitter. Most of these works classify the news as truth or fake by using the platform/user-specific information such as how popular the post is, the credibility of the user who shared it, diffusion patterns etc [LNL+15] [MGW+15]. Zhao, et al. [ZRM15] have used cue terms such as 'not true', 'unconfirmed' etc in retweets or the comments to detect fake news. The reasoning in their approach is that when people exposed to fake news they will comment or retweet with such words as a response to the post. Other studies focused on using the temporal characteristics of fake news during the spread. Kwon et al.[KCJ+13] used tweet volume in time series and Ma, et al.[MGW+15] measured variations of social context features over time.

All the research indicated before used datasets which are smaller in size. Wang [Wan17] curated dataset which consists approximately 13000 short statements by mining politfact.com covering a decade of information. In this approach, 6 machine learning models are built ranging from logistic regression to the convolutional neural network and compared. Along with the text data, metadata such as the speaker, subject, speaker history are also used. The convolutional neural network model used to capture surface level linguistics along with metadata performed better than other models.

All the attempts made in the above researches involve handcrafted feature engineering which is critical, biased and very time-consuming.

[3] https://www.nature.com
[4] https://www.britannica.com/

Jing Ma et al.[MGM+] efforts are focused on building a recurrent neural network (RNN) to detect rumors from Microblogs such as Twitter and Weibo effectively. The training dataset used is obtained by using constructed keywords in fake and truthy news from debunking services such as Snopes and Sina community management center. These keywords are used in Search API's of Microblogs and the tweet results of a search are labeled respectively. The social context information of a post and all its relevant posts such as comments or retweets is modeled as a variable-length time series. RNNs with different configurations such as one or two layers of GRU and LSTM are used and achieved very good results in capturing long-distance dependencies of temporal and textual representations of posts under supervision. This method completely avoids all the handcrafted feature engineering efforts which are biased and time-consuming. It produces better results with datasets from Twitter and Sina Weibo than all of the traditional Machine Learning methods. RNNs with two layers of GRU gave the best results and it was also very quick in predicting the rumor where the average time from debunking services.

The above method uses platform-specific features immensely such as retweets, comments in a tweet and the temporal correlation between them to figure out whether the news is true or fake. The features specified in one platform will be different from the other and it is not guaranteed that this methodology will give the same results for the same news in two different platforms. The semantics of the tweets are not used and it would remain the same across platforms.

Ciampaglia et al. [CSR+15] used DBPedia for checking computationally whether a given information is factual or not. The work uses the knowledge graph built from DBPedia which represents infobox section in Wikipedia. This represents only non-controversial and factual information which is analogous to human collected information. The methodology formulates the problem of checking facts into a network analysis problem which is finding the shortest path between nodes (subject and object of a sentence) in a graph. The aggregated generalities of nodes along a path in a weighted undirected graph are used as a metric for measuring the authenticity of information. The more the elements are generic; the weaker the truthfulness is. The genericness of a node is obtained by the degree of that node - no. of nodes connected to that node. The truthfulness of the information is improved if there exists at least one path from subject to an object with minimal non-generic nodes. This approach exploits the indirect connections to a great extent with distance constraints in a knowledge graph. The approach gave promising results when tested with datasets containing simple factual information about history, geography, entertainment, and biography.

The above research is a good initial step towards an automated fact checker system using the only semantics of data. The problem of fake news attempted is very primitive and uses only 'is' or 'type of' relation. The current fake news is very complex and subtle when it comes to ambiguities. In this approach, DBPedia is used and according to their sources, the update/synch frequency is slower than Wikipedia by 6 to 18 months.

3 Background Study

Neural networks are state of the art models to build learning systems. In the beginning, neural networks are inspired by the brain's computational mechanism [MP43]. But nowadays, it is also inspired from many applied mathematics such as linear algebra, probability, information theory and numerical optimization methods[GBC16]. Neural networks compose many interconnected fundamental functional units called neurons. Each neuron in the network takes in multiple scalar inputs and multiplies each input by a weight and then sums them, adds the result with a bias, applies a non-linear function at the end, which gives out a scalar output. There are different architectures of neural networks which vary mostly in how the neurons are connected to each other and how the weights are managed.

Feedforward neural networks [SKP97] can have multiple layers and each neuron in one layer is connected with every other neuron in the subsequent layer as given in the following figure.

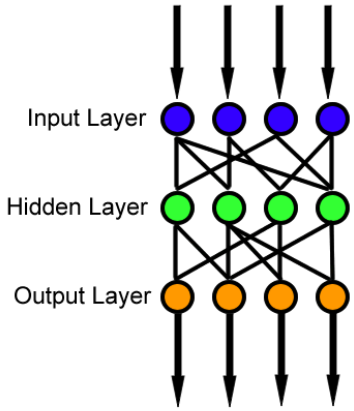

Figure 1: Feed forward neural network with 4 neurons each in every layer [Com06].

There are 3 layers in the figure and the extracted features of raw data will be sent through the input layer. Each circle is a neuron with incoming lines (Do not confuse with the arrows) as inputs and outgoing lines as outputs to the next layer. Each line carries a weight and the input layer has no weights since it has no incoming lines. The output layer has no outgoing lines and will be used as final output. In this figure, the hidden layers have 4 neurons which will have 4 weights and 1 bias variable for each of its neurons. In the beginning, the weights and bias variable are assigned with initial values. Each neuron can have a non-linear activation function such as sigmoid, hyperbolic tangent, rectifiers etc [Gol16]. Different activation functions pose different advantages and rectifiers are used in common. This activation function will help the neural network models to approximate any nonlinear function. The output layer can use a transformation function such as softmax to convert values to represent a discrete probability distribution. In this figure, 4 neurons are used and so it refers to 4 labels and this system classifies the input into one of the labels.

Training is an essential part of learning and like many supervised algorithms, a loss function is used to compute the error for the estimated output against the actual output. Some of the loss functions that could be used are hinge (binary and multiclass), log loss, categorical cross-entropy loss etc [Gol16]. The gradient of the errors is calculated and propagated back to compute with respect to weights and bias. The values of the weights and bias are adjusted with respect to the gradient and a learning parameter. Typically a random batch of inputs are selected and parameters are applied and the output is computed. The average loss is computed for that batch and the parameters are reassigned. This optimization technique is called stochastic gradient descent [Bot12] and other techniques available are Nesterov Momentum [SMDH13], AdaGrad [DHS11] etc. The overfitting in neural networks can be minimized by using regularization techniques such as L_2 regularization and dropout [HSK$^+$12]. The L_2 regularization works by adding a squared penalty on parameters with respect to the function being minimized. The dropout works by randomly ignoring half of the neurons in networks or in every specific layer in each batch and corrects the error only using the parameters of another half of neurons. This helps to prevent the network from relying on only specific weights.

Feedforward networks work very well on structured input data and in case of text data, the input is arbitrary. Techniques such as the continuous bag of words [MCCD13] can be used to convert the arbitrary input into fixed length but it will lose the order of the text which is important. Convolutional

neural network (CNN) [Ben97] are good at capturing the local characteristics of data irrespective of its position. In this, a nonlinear function is applied to every k-word sliding window and captures the important characteristics of the word in that window. All the important characteristics from each window are combined by either taking maximum or average value from each window. This captures the important characteristics of a sentence irrespective of their location. But yet the support for order is restricted only to local patterns and fails to recognize orders that are far apart in the sequence.

Recurrent neural networks (RNN) accept arbitrary input size, pays attention to the structure and consider the long dependencies [Elm]. RNN takes input as an ordered list of input vectors such as $x_{i:j}$ with initial state vector h_0 and returns an ordered list of state vectors $h_1,...,h_n$ as well as an ordered list of output vectors $o_1,...,o_n$. At time step t, RNN takes input a state vector h_{t-1}, an input vector x_t and outputs a new state vector h_t as shown in the figure. The outputted state vector is used as input state vector at the next time step. The same weights for input, state, and output vectors are used in each and every time step.

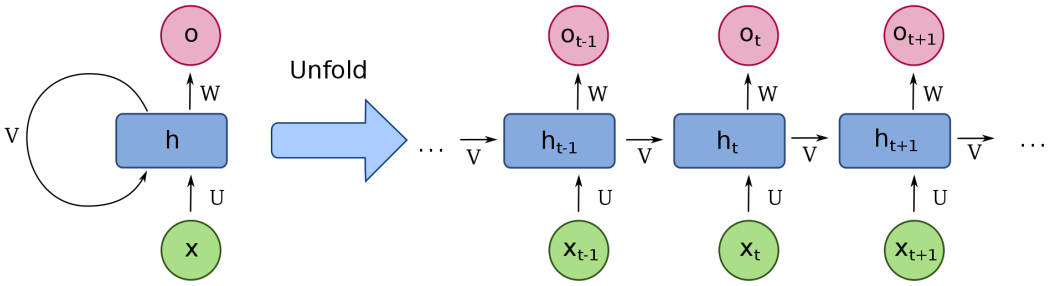

Figure 2: A basic example of RNN architecture [Del13].

To train RNN, the network is unrolled for the given input sequence and the loss function uses these nodes to compute the error and propagate backward depending on the application [Wer90]. While training, the error gradients might vanish or explode especially when dealing with RNNs. The gradient explosion can be handled by clipping the gradient when it goes beyond the threshold. LSTM networks [HS97] solves vanishing gradient problem by introducing memory cells which remembers gradients across time steps. These memory cells are controlled by mathematical functions which simulate logical gates called gating components. At each time step, a decision is made by gating components on how much of the current content of memory cell and new input should be retained.

4 Approach

Deep learning systems are giving better results in building intellectual systems nowadays.[Gol16] The results achieved in applications such as Autonomous car driving, playing chess are almost equivalent to the skillsets of a human. There are different neural network models exists such as Feed Forward, Convolutional, Recurrent Neural Network models. Wikipedia contains a lot of articles and each articles contains text. Text is a sequence data where as position of words in it is dependent on the other. Recurrent Neural Networks bring out best results in many applications involving text data such as machine translation, super tagging.

Every article or sentence in wikipedia can be fetched and inputted to the neural network as truthy value. But we lack false values. We need to use a semi supervised technique where in extraction of falsy values need to be carried out automatically. This is one of the challenges of this master thesis.

Some of the ideas in place are 1) Extract the articles and sentences from many of these fact checkers websites and use the ones which are labelled as false values. We need to make sure that they are opposites of sentences in wikipedia. 2) Build sentences which are opposite of sentences in wikipedia by using GLOVE technique or construct negative sentences. 3) Build sentences which are opposite of sentences in wikipedia by looking a semantic web representations. Word embeddings can be used to replace the verb opposites

The size of the article in wikipedia are long and arbitrary and RNN will face gradient diminishing problems. The long distance dependencies will be missed out and hence different configurations of the neural network should be used and compared 1) Different layers 2) Single/Multiple LSTM or GRU units 3) Usage of one hot vector vs word embeddings. The creation of word embeddings need to be thought through. It would be a good idea to do it from either wikipedia itself or from pre-trained word embeddiings

Recently convolutional neural network which character level input is giving out better results for some applications and this configuraiton should also be tried out.

Overall the things to do listed here are less and need to be appended more. Most of the stuff will be done in the prototyping phase itself . Will have to think about including ideas such as using DBPedia or any semantic database which is precisely expressed, Usage of any specific word embeddings or think of some other NLP techniques which could be used. Will have to be concrete at the same time. Need not be very strict - such as things specified here when both the parties agreed can be changed. Add new stuff, delete some other stuff and change or take a new direction. Things are possible but be reasonable

At some point when the methodology is specified, there should be a paragraph explaining about why these methodolgy are important and how different it is from the work done in related section. It can go here or somewhere in the introduction section

5 Evaluation

In order to do fake news classification using supervised machine learning approach, it is important to have a good and voluminous dataset. This dataset will be split and used for training, validation, and testing of the system. A general rule of thumb is to divide the dataset into 60-20-20 for training, validation, and testing respectively. But if the datasets are in order of millions then it is good enough to have approximately 10000 entries each for validation and testing. The validation dataset is used to compute the optimal values for the parameters of the model such as learning rate, no. of hidden layers, batch size etc.

It should be ensured that more or less equal amount of data should be present for both true and fake news. It is easy to extract true news for this project which is done by considering every statement in Wikipedia. Whereas curating fake news is a bit tricky since it does not exist.

Some of the approaches that could be taken to construct fake news are

1. Take each sentence from Wikipedia and distort it by swapping the words randomly.

2. There are two comprehensive open datasets available. One of it is published in Kaggle[5] which consists structured data of 13K size. The LIAR dataset is published by Wang [Wan17] which is mined from politifact.com and it consists approximately 13K entries.

3. Fake news can be obtained directly from facts checking websites such as snopes.com, politifact.com, etc., by crawling or using the API's provided.

The baseline system is a very simple RNN and the improvements are added on top of it. The following are the factors which are used to compare the baseline with the improved system

1. How good the system is able to classify the fake news?

2. How fast the model is build up?

The goodness of the system can be measured by metrics such as Precision, Recall, F1 measure and Accuracy. F1 measure is preferred since it includes both false positive and false negatives. The speed of system can be measured iteratively by calculating results of loss function. In this master thesis, the focus is mainly on improving the accuracy of the system and the speed of the system is handled by using good hardwares.

[5]https://www.kaggle.com/mrisdal/fake-news

6 Organizational matters

Duration of work:	01-July-2018 – 31-Dec-2018
Candidate:	Kandhasamy Rajasekaran
E-Mail:	kandhasamy@uni-koblenz.de
Student number:	216100855
Primary supervisor:	Prof. Dr. Steffen Staab
Supervisor:	Dr. Chandan Kumar
Secondary supervisor:	Lukas Schmelzeisen

7 Time schedule

- Introduction and Literature: 01-May-2018 – 30-June-2018
- Initial phase: 01-July-2018 – 15-Sep-2018
  - Prototyping: 01-July-2018 – 30-July-2018
  - ML pipeline implementation: 01-Aug-2018 – 15-Aug-2018
  - Baseline implementation: 16-Aug-2018 – 30-Aug-2018
  - Testing and refining: 01-Sep-2018 – 15-Sep-2018
- Development phase: 16-Sep-2018 – 30-Dec-2018
  - RNN with different configuration: 16-Sep-2018 – 30-Oct-2018
  - Comprehend benchmark results: 16-Oct-2018 – 21-Oct-2018
  - Analyse and figure out improvements: 22-Oct-2018 – 30-Oct-2018
  - Improvisation using NLP techniques: 01-Nov-2018 – 30-Dec-2018
  - Comprehend benchmark results: 16-Dec-2018 – 30-Dec-2018
- Final phase: 01-Jan-2019 – 01-Feb-2019
  - Comprehend Benchamark results: 01-Jan-2019 – 07-Jan-2019
  - Revision: 08-Jan-2019 – 22-Jan-2019
  - Thesis report: 01-Jan-2019 – 30-Jan-2019

A meeting with Lukas Schmelzeisen will happen approximately once in two weeks to discuss about the progress made and set the targets and milestones for subsequent weeks.

References

[AG17] Hunt Allcott and Matthew Gentzkow. Social Media and Fake News in the 2016 Election. *Journal of Economic Perspectives*, 2017.

[Ben97] Y Bengio. Convolutional Networks for Images, Speech, and Time-Series Parsing View project Oracle Performance for Visual Captioning View project. 1997.

[Bot12] Léon Bottou. Stochastic Gradient Descent Tricks. In Geneviève B. Montavon Grégoire and Orr and Müller Klaus-Robert, editors, *Neural Networks: Tricks of the Trade: Second Edition*, pages 421–436. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[Com06] Wikipedia Commons. Feed forward network, 2006. `https://en.wikipedia.org/wiki/File:Feed_forward_neural_net.gif`; CC BY-SA 3.0 (https://creativecommons.org/licenses/by-sa/3.0).

[CSR+15] Giovanni Luca Ciampaglia, Prashant Shiralkar, Luis M. Rocha, Johan Bollen, Filippo Menczer, and Alessandro Flammini. Computational fact checking from knowledge networks. *PLoS ONE*, 2015.

[Del13] Francois Deloche. Recurrent neural network unfold, 2013. [Online; accessed April 27, 2013; `https://commons.wikimedia.org/wiki/File:Recurrent_neural_network_unfold.svg`; CC BY-SA 4.0 (https://creativecommons.org/licenses/by-sa/4.0).

[DHS11] John Duchi, Elad Hazan, and Yoram Singer. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.

[Elm] Jeffrey L Elman. Finding Structure in Time. *COGNITIVE SCIENCE*, 14(1):179–21.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. www.deeplearningbook.org.

[Gol16] Yoav Goldberg. A Primer on Neural Network Models for Natural Language Processing. *Journal of Artificial Intelligence Research*, 57:345–420, 2016.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 1997.

[HSK+12] Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. 2012.

[KCJ+13] Sejeong Kwon, Meeyoung Cha, Kyomin Jung, Wei Chen, and Yajun Wang. Prominent features of rumor propagation in online social media. In *Proceedings - IEEE International Conference on Data Mining, ICDM*, 2013.

[LBB+18] David M. J. Lazer, Matthew A. Baum, Yochai Benkler, Adam J. Berinsky, Kelly M. Greenhill, Filippo Menczer, Miriam J. Metzger, Brendan Nyhan, Gordon Pennycook, David Rothschild, Michael Schudson, Steven A. Sloman, Cass R. Sunstein, Emily A. Thorson, Duncan J. Watts, and Jonathan L. Zittrain. The science of fake news. *Science*, 359(6380):1094–1096, 2018.

[LNL+15] Xiaomo Liu, Armineh Nourbakhsh, Quanzhi Li, Rui Fang, and Sameena Shah. Real-time Rumor Debunking on Twitter. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management - CIKM '15*, 2015.

[MCCD13] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.

[MGM+] Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha. Detecting Rumors from Microblogs with Recurrent Neural Networks.

[MGW+15] Jing Ma, Wei Gao, Zhongyu Wei, Yueming Lu, and Kam-Fai Wong. Detect Rumors Using Time Series of Social Context Information on Microblogging Websites. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management - CIKM '15*, 2015.

[MP43] Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 1943.

[PCL+07] Reid Priedhorsky, Jilin Chen, Shyong (Tony) K. Lam, Katherine Panciera, Loren Terveen, and John Riedl. Creating, destroying, and restoring value in wikipedia. In *Proceedings of the 2007 international ACM conference on Conference on supporting group work - GROUP '07*, 2007.

[SKP97] Daniel Svozil, Vladimir Kvasnieka, and Jie Pospichal. Chemometrics and intelligent laboratory systems Introduction to multi-layer feed-forward neural networks. *Chemometrics and Intelligent Laboratory Systems*, 39:43–62, 1997.

[SMDH13] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, (2010):8609–8613, 2013.

[VRA18] Soroush Vosoughi, Deb Roy, and Sinan Aral. The spread of true and false news online. *Science*, 359(6380):1146–1151, 2018.

[Wal05] Jimmy Wales. Internet encyclopaedias go head to head, 2005.

[Wan17] William Yang Wang. "Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection. 2017.

[Wer90] Paul J. Werbos. Backpropagation Through Time: What It Does and How to Do It. *Proceedings of the IEEE*, 1990.

[ZRM15] Zhe Zhao, Paul Resnick, and Qiaozhu Mei. Enquiring minds: Early detection of rumors in social media from enquiry posts. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, pages 1395–1405, Republic and Canton of Geneva, Switzerland, 2015. International World Wide Web Conferences Steering Committee.

8 Signatures

Kandhasamy Rajasekaran	Prof. Dr. Steffen Staab
Dr. Chandan Kumar	Lukas Schmelzeisen

9 Declaration of Authorship

I hereby declare that the thesis submitted is my own unaided work. All direct or indirect sources used are acknowledged as references.

I am aware that the thesis in digital form can be examined for the use of unauthorized aid and in order to determine whether the thesis as a whole or parts incorporated in it may be deemed as plagiarism. For the comparison of my work with existing sources I agree that it shall be entered in a database where it shall also remain after examination, to enable comparison with future theses submitted. Further rights of reproduction and usage, however, are not granted here.

This paper was not previously presented to another examination board and has not been published.

Koblenz, on July 10, 2018

_____

Kandhasamy Rajasekaran