Natural Language Processing with Assemblies of Neurons

Seung Je Jung

Georgia Institute of Technology

Undergraduate Thesis

December 17

Dr. Santosh Vempala – Mentor

Signature: DocuSigned by: 93D5B93B1A7C4F9...

Date: 12/17/2021 | 2:05 PM EST

Dr. Gerandy Brito – Second Reader:

Signature: DocuSigned by: Gerandy Brito F9E075EABAF442F...

Date: 12/17/2021 | 7:54 AM EST

Abstract

The Assembly Calculus is a novel framework intended to bridge the gap between the level of neuron and synapses, and that of cognition. The Assembly Calculus is a computational system entailing (18) a basic data item called an assembly, a stable set of neurons explained below; (14) a set of operations that create and manipulate assemblies; and (19) an execution model which is squarely based on basic tenets of neuroscience. Importantly, it allows the creation of biologically plausible, flexible and interpretable programs, enabling one to develop tangible hypotheses on how specific brain functions may work. Here, to help lay groundwork for the creation of algorithms in this framework, we present a natural language processing algorithm to solve the analogy task. Further, to facilitate such experimentation, we present here a tool which in real-time allows the simulation, modification and visualisation of this computational system, including several prepared examples. Lastly, we also present empirical analysis of the capabilities of the assembly calculus to store information in brain areas.

Introduction

Over the past decade, natural language processing (NLP) has seen incredible gains in performance, with GPT-3 by Open-AI recently becoming able to create news articles which humans were unable to distinguish from human written ones (9, 10). Most of these gains have come from training increasingly large models and architectural improvements such as encoder-decoder networks and transformers (8, 10). However, although these networks may achieve incredible performance, they are incredibly rigid, only being able to complete the task they were created for. Furthermore, they are black boxes in the sense that we cannot observe how the model works, and so these networks cannot provide any insights into how humans process language.

To develop a more generalisable NLP model, a newly developed model of computation can be utilized, which mimics low level brain structures (14). The model is comprised of a finite number of brain areas, each of which contains a fixed number of neurons, with neurons in the same area having a fixed probability of being connected to each-other (13). Between certain areas, neurons are permitted to have connections to those in the other area. Neurons fire based on incoming connections, and at each timestep, each area can only have a set number of neurons firing, with neurons with higher activations being chosen over ones with lower activations. These neurons also obey Hebbian plasticity, so if a neuron A which is connected to B fires in time step t, and in the next timestep the neuron B fires, the connection is strengthened and becomes more likely to fire in the future (14). These mechanisms form densely interconnected areas named assemblies, analogous to densely connected areas in biological brains which represent arbitrary objects such as concepts or words (14). Using this model, we can perform operations on these assemblies including projection (copying) and association (combine two assemblies) to perform computation.

Unlike DL architectures, we can observe how the model learns and performs tasks since we can control the operations in the model and watch a visualisation of the model change over time (14). The changes we observe are unlike gradients which are uninterpretable, and are instead very intuitive changes such as copying and strengthening connections. In addition to this,the model's increased flexibility in permissible operations allows better generalisation to different tasks.

As a proof of concept to show the model's viability in NLP tasks and to provide a base model for other tasks, this paper will describe a model based on the brain computation framework to solve analogy tasks such as "'man' is to 'king' as 'woman' is to '?'" where the model predicts the best word to complete the analogy (2). Additionally, we present a web-app, which was presented at NeurIPS 2021 demos, used to visualise and modify the model so that we can easily evaluate and observe the inner workings of the model (16, 17). We then present experimental results to verify how well the model performs under different constraints to establish bounds on brain computation. We hope that this research will provide a pathway for us to create increasingly sophisticated models which offer flexibility advantages over deep learning models, as well as improve our understanding of the brain.

Literature Review

Natural language processing has recently been a popular field of research, and has progressed to the point of reaching human human performance in many tasks including text synthesis (10). Most of this progress has been achieved through increasingly complex deep learning models as well as newer architectures. However, these are restricted by the fact that deep learning networks are rigid and highly specialised for tasks. By instead using a model of structures in the human brain, we will try to build an interpretable and more generalisable way of solving NLP tasks. Specifically, we will try to solve the word analogy problem which attempts to produce the best fitting word given the structure "Man is to King as woman is to …". We aim to produce a basis architecture which future architectures can build upon to take advantage of the more general brain assembly model which we base our model on.

The first issue in NLP is how to encode the data for input into the neural network. The most straightforward idea would be to one-hot encode the words, however, this encoding lacks any representation of what the words actually mean, and also poses the problem of necessitating a very high dimensional input space. A better way to represent words is to train a one layer neural network originally with a one hot encoding to guess a missing word in a sentence, and then use the weights from the one layer neural network as the vector for a word (1). This word embedding, one form of Word2Vec, produces good results, encoding not only words with similar meanings into similar areas, but also encoding relationships between words (1). Research into the optimality of such word embeddings has also been conducted, including results demonstrating what the optimal dimensionality of the embedding to use would be (15), since having too large a dimensionality creates issues with scalability of a model as well as overfitting, whereas a smaller dimensionality may not capture the differences between words.

The first network able to process sequential data meaningfully was recurrent neural networks (RNN), which were conceived as networks having cyclic connections, in contrast to only forward connections (3). This network structure was the first to allow the propagation of data from previous inputs, which enables a notion of memory carried over from the previous inputs (4). A large issue with RNNs, however, is that long term dependencies are not well retained since they must propagate through gradients multiple cycles. Intuitively, if the gradient were to be less than one, raising it to a large power would cause it to vanish, whereas if the gradient were greater than one it would diverge to infinity. Thus, long short term memory networks (LSTMs) were conceived, which have a dedicated line through which memory is sent through to the next recurrence (4). Since the data is separately passed through in a more controlled manner, a lot of the impact of vanishing and exploding gradients is mitigated, but the problem still exists in the sense that extremely long term dependencies are still less likely to be retained through cycles.

Another issue that LSTMs suffer from is that the input and output sequence lengths are fixed which for language tasks, such as translation, is inadequate since translated sentences are not necessarily always the same length. This issue was overcome by the conceptualisation of the encoder-decoder network, which uses one encoder LSTM to produce an fixed length information vector, and a decoder LSTM to read the information vector and produce a sequence as the network output (6). The information vector produced becomes an abstraction for what is important in the input sentence, and with the new flexibility in the lengths of the inputs and outputs, state of the art performance was achieved in translation tasks with even small networks (7). Since the network utilised LSTMs, however, it still was restricted by the long term dependency issue in LSTMs.

The general concept of attention based networks was first conceptualised in a translation task model, with each word having a matrix representing its relation to other words in the sentence (5). These matrices fundamentally acted akin to how humans pay special attention to specific words in a sentence when performing language tasks. Using this architecture along with a bidirectional LSTM, the authors were able to again achieve state of the art performance in machine translation tasks.

One of the most influential architectures in NLP, the transformer, was introduced which combined the ideas of attention based networks and encoder decoder networks and finally resolved the long standing issue of long term dependencies. In the transformer architecture, there is no longer any notion of memory, and vectors simply encode what other words to attend to (8). The transformer architecture is comprised similarly of a stack of encoders and decoders, with each encoder and decoder having a multi-head attention layer which contain dot product attention vectors which implement the attention mechanism (8). In addition to removing the long term dependency problem, LSTMs take long amounts of time to train since the gradients have to be backpropagated sequentially for every word, but for transformers since there is no recurrence, the entire network can be trained in a massively parallel fashion, allowing for very fast training times (8). Almost all newer, state of the art architectures such as GPT-3, BERT and roBERTa utilise solely a transformer architecture, usually with a massive number of weights due to how efficient the training is, and how well transformer architectures perform (9, 10, 11). In fact, word embeddings have become obsolete in favour of specialising BERT for each task instead.

However, this is not to say word embeddings have no use. The analogy task for NLP consists of a puzzle of the shape A:B::C:D which represents the analogy "A is to B as C is to D". The purpose of the task is to try to guess what the best fit for D would be. GPT-3 and fine-tuned BERT both perform well on this task, nearing state of the art performance (9, 10), however, word embeddings, due to their nature of encoding the relationships between words can in a simple manner also solve analogy tasks well (2).

The issue with all of these solutions and networks is that they are all completely rigid and have to be retrained from the ground up upon a change in the network structure. Although initially conceptualised as mimicking neuron structure in the brain, deep learning does not reflect a sophisticated model of the brain. Furthermore, it would be difficult to argue that deep learning truly understands the abstractions that we make, and also the information learned in one field cannot easily be generalised to other fields like humans can. To combat this problem, a new architecture for modelling neurons in the brain was introduced utilising Hebbian plasticity and neuron assemblies (13, 14). With the new architecture, dense areas of neurons can be arbitrarily connected to each other and modified in a manner that is biologically plausible, and one that mimics much more accurately than deep learning how human brains might think. In the quest to both better understand the human brain's workings and produce human level generalisation skills, this model may be much more fit.

Method

Analogy Model

We first ran investigations on various machine learning approaches to analogy tasks, and decided on a design where we had 5 separate brain areas: Vocab Area 1, Vocab Area 2, Word Pair Area, Relation Area and the Output Area. We also had training data of the form (Word A, Word B, Relation) which describes that Word A is related to Word B by a relation Relation, such as (Black, White, Opposite).

The vocab areas each contain assemblies for every word that is mentioned in the training data, and this was achieved by projecting a stimulus corresponding to each word into the respective vocab areas 10 times. The word pair area contains every pair of words related to each-other in the training data, and this was achieved similarly by projection. For each relation (A, B, R), we project stimulus A into Vocab Area 1, and stimulus B into Vocab Area 2 in order to prime the relevant assemblies. Then, we simultaneously project Vocab Area 1 and Vocab Area 2 into the Word Pair Area simultaneously to associate the resultant word pair in the Word Pair Area. We also swap the ordering of the words in order to associate both orderings (A, B, R) and (B, A, R). The purpose of the relation area is to identify which relation each word pair has. To do this, we prime a certain word pair by following the same process as above, and then we project from the Word Pair Area into the Relation Area together with a stimulus which represents the relation. Then, we have the output area which will output a word pair representing the answer to the analogy puzzle. To achieve this, we iterate over all pairs of words and their relations, and first project word A into Vocab Area 1 project R into the Relation Area to prime the assemblies. Then, we will project simultaneously from Vocab Area 1 and the Relation Area to the Output Area, simultaneously with a stimulus which represents the word pair AB.

After this training, we can use the model to solve the analogy puzzle. For the puzzle A:B::C:?, we will first project A into Vocab Area 1 and B into Vocab Area 2. This will activate the respective assemblies in the areas. Then, we simultaneously project from Vocab Area 1 and Vocab Area 2 into the Word Pair Area, which will cause the relevant word pair assembly to fire as it will have the highest activation. We can then project the Word Pair Area into the Relation Area in order to determine what relation the word pairs have, since we will have learned this during training. Lastly, we project C into Vocab Area 1, and project both Vocab Area 1 and the Relation Area into the Output Area simultaneously in order to get the answer.

This was designed and implemented in a Python reference implementation of the assembly calculus to guarantee correctness, and we evaluated the model to have >90% correctness on a small dataset. Then, we moved to a C++ implementation for greater speed to allow for testing larger datasets. We determined with larger scale testing that there were issues with older assemblies drifting as we projected more relations over time, and accordingly adjusted the model. Specifically, we decreased plasticity and projected relations over multiple rounds in order to more strongly form them, and keep assemblies from drifting too far while other assemblies are being created. After this, we were able to achieve comparable accuracy on hundreds of word pairs.

Simulation and Visualization Web App

The web app was designed to allow neuroscientists who are unfamiliar with computer science to easily create programs with, and understand visualizations of their programs. To achieve this goal, we adopted a scratch-like interface to create programs with. On the left panel of Figure 1, is the toolbox tab where users can select a set of blocks to choose from. In the middle-left is the workspace where users can create algorithms in. On the right side the user can run the program using the Run Program button and see the visualization of their algorithm at different points in time by moving the slider forward and back. Pre-packaged examples are provided for the user to familiarize themselves with the interface and include the operations Project and Associate, and sample programs including the earlier mentioned NLP analogy algorithm. The web-app was submitted and accepted to NeurIPS 2021 demos where it was presented.

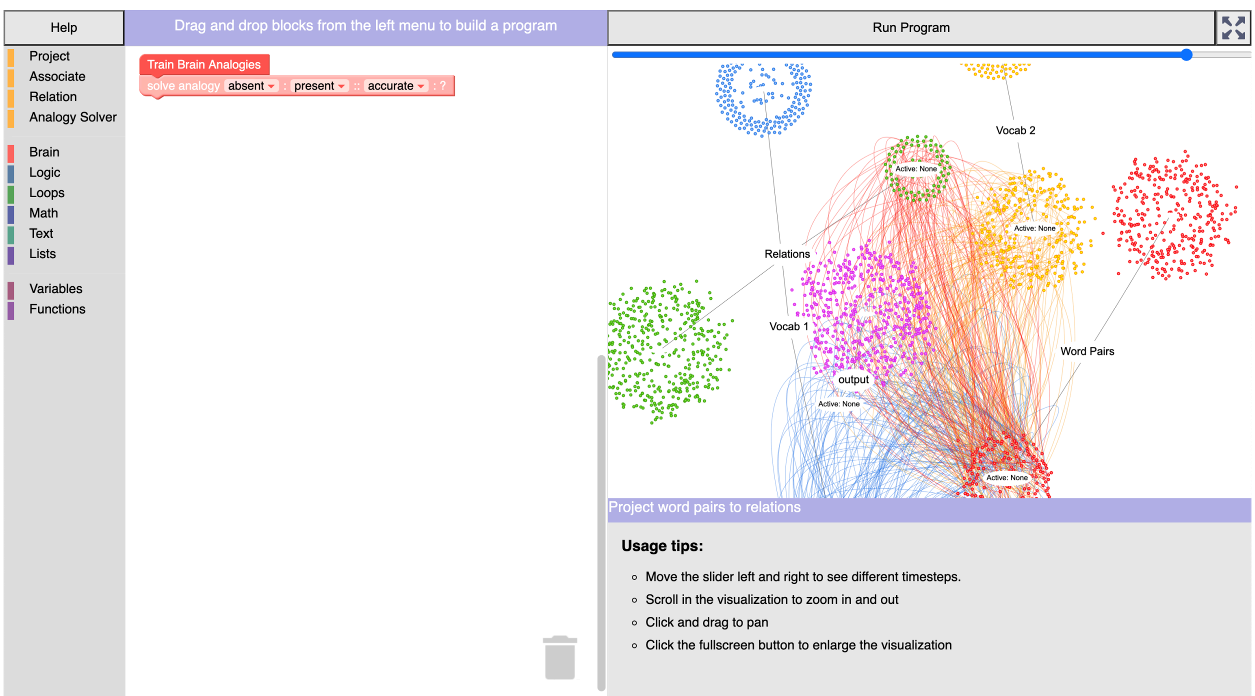

Figure 1

Experimental Analysis of Capacity

To establish some initial bounds on the assembly capacity of areas, we designed and ran experiments. Our experimental design seeked to find out for an area of a fixed size, n, with k = sqrt(n), what is the maximum number of assemblies which can be supported before less than 90% of them can be recalled correctly. In order to do this, we created two areas, an input area and a memory area. To create an assembly in the memory area, we chose k neurons from the input area and projected them to the memory area. We then turned off plasticity, chose a random sample of up to 50 of these created assemblies, fired the same k neurons from the input area and then checked which assembly fired in the output area. If the assembly fired was different from the assembly initially formed, then we can conclude that there was an issue. If we find that with a 90% confidence interval, more than 90% of the assemblies would be correctly recalled, then we added a new assembly until this was no longer true. We then repeated the same experiment with a larger n, and averaged over 50 trials for each n. The raw experimental results can be seen in Figure 2, with a log scale on Figure 3. On Figure 3, we can see that the gradient of the line past n = 750 is greater than 1, suggesting a super-linear relationship between an increase in n and an increase in the capacity.

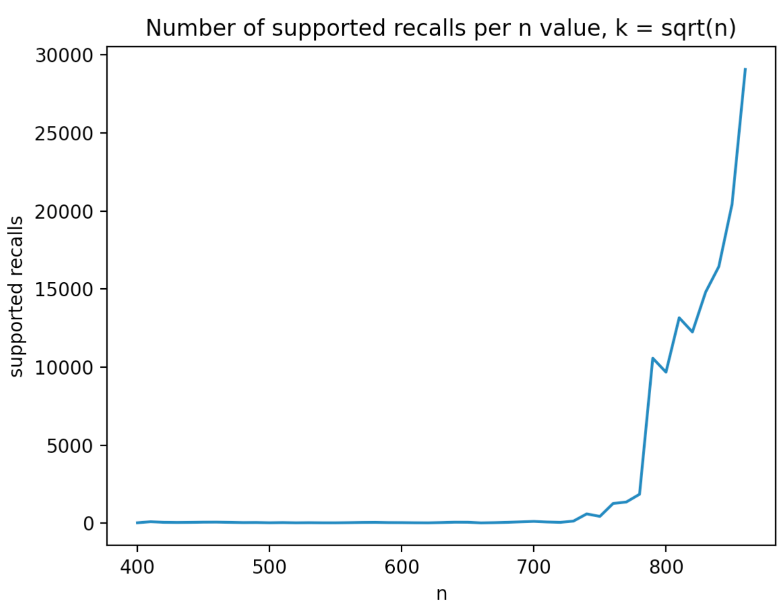

Figure 2

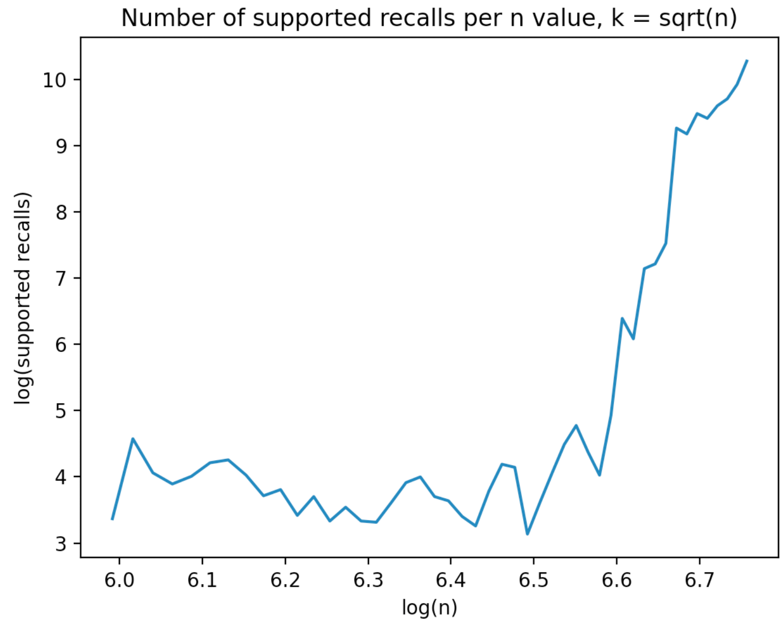

Figure 3

Conclusion

In this paper we have laid foundation for creating more natural language processing algorithms by designing an algorithm which solves the analogy problem of "man is to King as woman is to ?". We successfully achieved accuracies of 90% on several hundred word pairs. Further, we developed a web application with a simple interface, which can be used by neuroscientists and computer scientists to design and gain insights into their algorithms and the assembly calculus. Lastly, we found through empirical experiments that the capacity of brain areas increases super-linearly with respect to the number of neurons in it. Further work into this area could involve creating NLP models which do not require relation input, and can learn from just reading like GPT-3 or BERT. Additionally, the web-app could be improved to include more examples and have additional blocks added to make creating programs simpler so that it is more accessibly for users. Finally, experiments at a larger scale would give more evidence for super-linear scaling, or theoretical lower bounding on the capacity could help this as well.

References

1. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.
2. Bolukbasi, T., Chang, K. W., Zou, J., Saligrama, V., & Kalai, A. (2016). Man is to computer programmer as woman is to homemaker? debiasing word embeddings. arXiv preprint arXiv:1607.06520.
3. Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). Learning internal representations by error propagation. California Univ San Diego La Jolla Inst for Cognitive Science.
4. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.
5. Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.
6. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.
7. Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. arXiv preprint arXiv:1409.3215.
8. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. arXiv preprint arXiv:1706.03762.
9. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
10. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. arXiv preprint arXiv:2005.14165.
11. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

12. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., ... & Bengio, Y. (2015, June). Show, attend and tell: Neural image caption generation with visual attention. In International conference on machine learning (pp. 2048-2057). PMLR.
13. Papadimitriou, C. H., & Vempala, S. S. Random Projection in the Brain and Computation.
14. Papadimitriou, C. H., Vempala, S. S., Mitropolsky, D., Collins, M., & Maass, W. (2020). Brain computation by assemblies of neurons. Proceedings of the National Academy of Sciences, 117(25), 14464-14472.
15. Yin, Z., & Shen, Y. (2018). On the dimensionality of word embedding. arXiv preprint arXiv:1812.04224.
16. Jung, S. J., & Vempala, S. S. (n.d.). Brain Computation Simulation. http://braincomputationsimulation.cc.gatech.edu/brain.
17. Papadimitriou, C. H., Vempala, S. S., Mitropolsky, D., Collins, M., & Maass, W. (n.d.). *Code for simulating assembly operations*. GitHub. https://github.com/santoshv/assemblies.
18. Buzsáki, György(2010): *Neural syntax: cell assemblies, synapsembles, and readers*, 3: 362–385.
19. Mitropolsky, Daniel / Collins, Michael J / Papadimitriou, Christos H(2021): *A Biologically Plausible Parser.*

DocuSign

Certificate Of Completion

Envelope Id: EC62341D75184B9BAE09437AFF4969AF		Status: Completed
Subject: Please DocuSign: Thesis.pdf		
Source Envelope:		
Document Pages: 13	Signatures: 2	Envelope Originator:
Certificate Pages: 5	Initials: 0	Seung Je Jung
AutoNav: Enabled		North Avenue
EnvelopeId Stamping: Disabled		Atlanta, GA 30332
Time Zone: (UTC-05:00) Eastern Time (US & Canada)		sjung323@gatech.edu
		IP Address: 73.137.193.153

Record Tracking

Status: Original 12/17/2021 12:38:02 AM	Holder: Seung Je Jung sjung323@gatech.edu	Location: DocuSign

Signer Events	Signature	Timestamp
Gerandy Brito gbrito3@gatech.edu Security Level: Email, Account Authentication (Optional)	DocuSigned by: Gerandy Brito F9E075EABAF442F... Signature Adoption: Pre-selected Style Using IP Address: 136.55.36.49	Sent: 12/17/2021 12:39:25 AM Viewed: 12/17/2021 7:53:33 AM Signed: 12/17/2021 7:54:01 AM

Electronic Record and Signature Disclosure:
Accepted: 12/17/2021 7:53:33 AM
ID: bb6d6578-2dfd-4d6e-b1ba-7fa240f69e52

Santosh Vempala svempala3@gatech.edu Security Level: Email, Account Authentication (Optional)	DocuSigned by: 93D5B93B1A7C4F9... Signature Adoption: Drawn on Device Using IP Address: 128.61.37.109	Sent: 12/17/2021 12:39:25 AM Viewed: 12/17/2021 2:05:03 PM Signed: 12/17/2021 2:05:24 PM

Electronic Record and Signature Disclosure:
Accepted: 12/17/2021 2:05:03 PM
ID: 0f0ea7f4-ab2c-436a-a9c8-56d6ecf3fb44

In Person Signer Events	Signature	Timestamp

Editor Delivery Events	Status	Timestamp

Agent Delivery Events	Status	Timestamp

Intermediary Delivery Events	Status	Timestamp

Certified Delivery Events	Status	Timestamp

Carbon Copy Events	Status	Timestamp

Witness Events	Signature	Timestamp

Notary Events	Signature	Timestamp

Envelope Summary Events	Status	Timestamps
Envelope Sent	Hashed/Encrypted	12/17/2021 12:39:25 AM
Certified Delivered	Security Checked	12/17/2021 2:05:03 PM

Envelope Summary Events	Status	Timestamps
Signing Complete	Security Checked	12/17/2021 2:05:24 PM
Completed	Security Checked	12/17/2021 2:05:24 PM

Payment Events	Status	Timestamps

Electronic Record and Signature Disclosure

ELECTRONIC RECORD AND SIGNATURE DISCLOSURE

From time to time, Georgia Institute of Technology (we, us or Company) may be required by law to provide to you certain written notices or disclosures. Described below are the terms and conditions for providing to you such notices and disclosures electronically through the DocuSign system. Please read the information below carefully and thoroughly, and if you can access this information electronically to your satisfaction and agree to this Electronic Record and Signature Disclosure (ERSD), please confirm your agreement by selecting the check-box next to 'I agree to use electronic records and signatures' before clicking 'CONTINUE' within the DocuSign system.

Getting paper copies

At any time, you may request from us a paper copy of any record provided or made available electronically to you by us. You will have the ability to download and print documents we send to you through the DocuSign system during and immediately after the signing session and, if you elect to create a DocuSign account, you may access the documents for a limited period of time (usually 30 days) after such documents are first sent to you. After such time, if you wish for us to send you paper copies of any such documents from our office to you, you will be charged a $0.00 per-page fee. You may request delivery of such paper copies from us by following the procedure described below.

Withdrawing your consent

If you decide to receive notices and disclosures from us electronically, you may at any time change your mind and tell us that thereafter you want to receive required notices and disclosures only in paper format. How you must inform us of your decision to receive future notices and disclosure in paper format and withdraw your consent to receive notices and disclosures electronically is described below.

Consequences of changing your mind

If you elect to receive required notices and disclosures only in paper format, it will slow the speed at which we can complete certain steps in transactions with you and delivering services to you because we will need first to send the required notices or disclosures to you in paper format, and then wait until we receive back from you your acknowledgment of your receipt of such paper notices or disclosures. Further, you will no longer be able to use the DocuSign system to receive required notices and consents electronically from us or to sign electronically documents from us.

All notices and disclosures will be sent to you electronically

Unless you tell us otherwise in accordance with the procedures described herein, we will provide electronically to you through the DocuSign system all required notices, disclosures, authorizations, acknowledgements, and other documents that are required to be provided or made available to you during the course of our relationship with you. To reduce the chance of you inadvertently not receiving any notice or disclosure, we prefer to provide all of the required notices and disclosures to you by the same method and to the same address that you have given us. Thus, you can receive all the disclosures and notices electronically or in paper format through the paper mail delivery system. If you do not agree with this process, please let us know as described below. Please also see the paragraph immediately above that describes the consequences of your electing not to receive delivery of the notices and disclosures electronically from us.

How to contact Georgia Institute of Technology:

You may contact us to let us know of your changes as to how we may contact you electronically, to request paper copies of certain information from us, and to withdraw your prior consent to receive notices and disclosures electronically as follows:
To contact us by email send messages to: eis-distributed-applications@oit.gatech.edu

To advise Georgia Institute of Technology of your new email address

To let us know of a change in your email address where we should send notices and disclosures electronically to you, you must send an email message to us at docusign_admins@lists.gatech.edu and in the body of such request you must state: your previous email address, your new email address. We do not require any other information from you to change your email address.

If you created a DocuSign account, you may update it with your new email address through your account preferences.

To request paper copies from Georgia Institute of Technology

To request delivery from us of paper copies of the notices and disclosures previously provided by us to you electronically, you must send us an email to eis-distributed-applications@oit.gatech.edu and in the body of such request you must state your email address, full name, mailing address, and telephone number. We will bill you for any fees at that time, if any.

To withdraw your consent with Georgia Institute of Technology

To inform us that you no longer wish to receive future notices and disclosures in electronic format you may:

i. decline to sign a document from within your signing session, and on the subsequent page, select the check-box indicating you wish to withdraw your consent, or you may;

ii. send us an email to eis-distributed-applications@oit.gatech.edu and in the body of such request you must state your email, full name, mailing address, and telephone number. We do not need any other information from you to withdraw consent.. The consequences of your withdrawing consent for online documents will be that transactions may take a longer time to process..

Required hardware and software

The minimum system requirements for using the DocuSign system may change over time. The current system requirements are found here: https://support.docusign.com/guides/signer-guide-signing-system-requirements.

Acknowledging your access and consent to receive and sign documents electronically

To confirm to us that you can access this information electronically, which will be similar to other electronic notices and disclosures that we will provide to you, please confirm that you have read this ERSD, and (i) that you are able to print on paper or electronically save this ERSD for your future reference and access; or (ii) that you are able to email this ERSD to an email address where you will be able to print on paper or save it for your future reference and access. Further, if you consent to receiving notices and disclosures exclusively in electronic format as described herein, then select the check-box next to 'I agree to use electronic records and signatures' before clicking 'CONTINUE' within the DocuSign system.

By selecting the check-box next to 'I agree to use electronic records and signatures', you confirm that:

- You can access and read this Electronic Record and Signature Disclosure; and
- You can print on paper this Electronic Record and Signature Disclosure, or save or send this Electronic Record and Disclosure to a location where you can print it, for future reference and access; and
- Until or unless you notify Georgia Institute of Technology as described above, you consent to receive exclusively through electronic means all notices, disclosures, authorizations, acknowledgements, and other documents that are required to be provided or made available to you by Georgia Institute of Technology during the course of your relationship with Georgia Institute of Technology.