ATENEO DE MANILA
UNIVERSITY
Loyola Schools

A. COURSE INFORMATION

COURSE NUMBER	MATH 103.1	**NO. OF UNITS**	3
COURSE TITLE	Predictive Modeling for Text		
PREREQUISITE/S	MATH 71.1 Fundamentals of Computing I		
DEPARTMENT	Mathematics	**SCHOOL**	Science and Engineering
SCHOOL YEAR	2022-2023	**SEMESTER**	1st
INSTRUCTOR/S	Felix P. Muga II, Ph.D.		
VENUE/PLATFORM	On-Site/Canvas	**SECTION**	B
SCHEDULE	Monday and Thursday, 09:30-11:00 A.M, 11 August – 10 December 2022		

B. COURSE DESCRIPTION

This course introduces the students to the supervised or predictive modeling for text. It covers the processes in transforming text into data formats that are useful for modeling. It discusses machine learning algorithms and puts them into practice to predict outcomes using these transformed text data. Lastly, it includes some deep learning models for text to do the same tasks and have the same goals as the machine learning models.

WHERE IS THE COURSE SITUATED WITHIN THE FORMATION STAGES IN THE FRAMEWORK OF THE LOYOLA SCHOOLS CURRICULA	
	FOUNDATIONS: Exploring and Equipping the Self
✓	ROOTEDNESS: Investigating and Knowing the World
	DEEPENING: Defining the Self in the World
	LEADERSHIP: Engaging and Transforming the World

C. PROGRAM LEARNING OUTCOMES

Alignment of Program to the Core Curriculum Learning Outcomes

The Ideal Ateneo Graduate: A Person of Conscience Competence Compassion Commitment							
CCLO 1	CCLO 2	CCLO 3	CCLO 4	CCLO 5	CCLO 6	CCLO 7	CCLO 8
	CLO 1-6						

Alignment of the Course to the Program Learning Outcomes

BS Mathematics

MCLO1	MCLO2	MCLO3	MCLO4	MCLO5	MCLO6	MCLO7	MCLO8	MCLO9	MCLO10
		✓		✓	✓	✓		✓	

BS Applied Mathematics (Financial Mathematics and Data Science)

MCLO 1	MCLO 2	MCLO 3	MCLO 4	MCLO 5	MCLO 6	MCLO 7	MCLO 8	MCLO 9	MCLO 10	MCLO 11	MCLO 12
			✓	✓	✓	✓		✓	✓	✓	✓

D. COURSE LEARNING OUTCOMES

Alignment of the Course to the Core Curriculum Learning Outcomes

At the end of the course, the students should be able to:

COURSE COMPETENCIES AND COURSE LEARNING OUTCOMES
COMPETENCE 1 Preprocess text data as a useful and efficient format for predictive modeling • **CLO1** (**Knowledge**) Explain the framework on text data preprocessing • **CL02** (**Skill**) Perform the different steps in preprocessing text data like tokenization, normalization, and substitution. • **CLO3 (Attitude)** Appreciate the technique of preparing and organizing text into a tidy data format.

COMPETENCE 2 Learn how to use transformed text data for predictive modeling

- **CLO4 (Knowledge**) Identify the different methods in developing predictive models from transformed text data
- **CLO5 (Skill**) Perform the different methods in predictive modeling with text data
- **CLO6** Assess the value of the different methods in predictive modeling from transformed text data

C. COURSE OUTLINE and LEARNING HOURS

Course Outline	CLOs	Estimated Contact or Learning Hours
Data Exploration and Data Wrangling	CLO 1	5
Improving Programming Skills	CLO 1, 2	5
Data Modeling	CLO 1, 2, 3	5
Tokenization and Stopwords	CLO 1, 2, 3	5
Stemming and Word Embeddings	CLO 1, 2, 3	5
Regression	CLO 4, 5, 6	5
Classification	CLO 4, 5, 6	5
Dense Neural Networks	CLO 4, 5, 6	5
Long Short-Term Memory	CLO 4, 5, 6	5
Convolutional Neural Networks	CLO 4, 5, 6	5
		50

D. ASSESSMENTS

Assessment Tasks	Date Due	Assessment Weight	CLOs
Long Test 1 (100 points)	Sept 05 (Mon)	20%	CLO 1, 2, 3
Long Test 2 (100 points)	Oct 03 (Mon)	20%	CLO 1, 2, 3
Long Test 3 (100 points)	Nov 07 (Mon)	20%	CLO 4, 5, 6
Check for Understanding (100 points)	TBA per Module	20%	CLO 4, 5, 6
Finals: Project (100 points)	Dec 8 (Thu) and Dec 10 (Sat)	20%	CLO 4, 5, 6

F. TEACHING and LEARNING METHODS

TEACHING & LEARNING METHODS and ACTIVITIES	CLOs
On-Site and Synchronous Online Sessions via Zoom	CLO 1, 2, 3, 4, 5, 6
Asynchronous Solving of Exercises and Check for Understanding	CLO 1, 2, 3, 4, 5, 6
Consultations	CLO 1, 2, 3, 4, 5, 6
Guided Self-Study	CLO 1, 2, 3, 4, 5, 6

G. REQUIRED READINGS

1. Wickham, Hadley and Garrett Grolemund. ***R for Data Science*** : ***Import, Tidy, Transform, Visualize, and Model Data*** (physical copy) : O'Reilly Media; 1st edition (January 17, 2017); (free website version, last built 2021-04-29) : https://r4ds.had.co.nz/index.html

2. Hvitfeldt, Emil and Julia S.,. ***Supervised Machine Learning for Text Analysis in R*** (physical copy) : Chapman and Hall/CRC, 1st edition, October 22, 2021 (Forthcoming) (free website version, last built on 2021-07-28) https://smltar.com/

H. SUGGESTED READINGS
(See Appendix)

I. GRADING SYSTEM

Total Number of Points	500 points total

Final Grade (FG)	Letter Grade	Final Grade	Letter Grade
FG ≥ 92	**A**	**60 ≤ FG < 68**	**C**
86 ≤ FG < 92	**B+**	**50 ≤ FG < 59**	**D**
77 ≤ FG < 86	**B**	**FG < 50**	**F**
69 ≤ FG < 76	**C+**		

J. CLASS POLICIES

GENERAL GUIDELINES/HOUSE RULES

1. The topics for this course are organized into **modules** that are uploaded in the MATH 51.3 Canvas Learning Management System (LMS), which we refer here as our ***course site***.

2. The modules will be made available on or before the listed schedule in our course site. In addition to the lectures during onsite and online synchronous classes, students are expected to read the discussions in the course site, read the uploaded slides, watch videos, and visit some websites. Open the "Resources" section of our course site regularly. A student with unstable or no internet connection may avail of the **portable learning packet** (PLP) that can be requested through LS One (www.ateneo.edu/lsone).

3. The learning modules in the course site are based on our **textbooks** (listed as a required textbook in this syllabus). Instructions on how to get a copy of the book is posted in the course site. It is highly recommended that in addition to the module, students read the discussions on the textbooks and answer exercises at the end of the section. Suggested exercises will be included in the module.

4. Materials posted on our course site on Canvas **must not be shared** in any other website and social media and with other students not enrolled in the course. The University is committed to ensuring that copyrighted works of the creators are respected and protected against misuse and unauthorized use. For further information, visit: https://www.aipo.ateneo.edu/ippolicy and https://www.aipo.ateneo.edu/copyright-guidelines

5. If you have concerns, questions, and clarifications about the lessons, you may schedule a **small group consultation** during the instructor's consultation time given below or by appointment. Consultations maybe on campus or via Zoom. Contact the instructor, **during office hours**, using the messaging facility of Canvas or email for any concerns related to the course.

6. Students are expected to exercise the highest level of **academic integrity**. Students are expected to treat their classmates and instructor, **with respect** at all times even in an online environment. Cheating, plagiarism, discourtesy or misbehavior will not be tolerated and will be treated in accordance to the Student Handbook. The class adheres to the LS Gender Policy: http://www.ateneo.edu/ls/ls-gender-policy. The class also adheres to the Code of Decorum and Administrative Rules on Sexual Harassment, Other Forms of Sexual Misconduct, and Inappropriate Behavior: https://www.ateneo.edu/policies/code-decorum-investigation-sexual-harassment.

ON-SITE AND ONLINE SYNCHRONOUS CLASSES

1. Majority of our classes will be held **Onsite**. All long tests will also be held Onsite. There will also be lectures and problem-solving **Synchronous Sessions** via Zoom, during our class, usually on Thursdays. The schedule is given below, and the Zoom link is posted in the "Home" page on Canvas. Attendance will be checked.

2. Before the On-site and Online Sync Session, students are expected to **study** all topics in the module that are related to or pre-requisite to the topic of the session. Be ready with your questions for clarification.

3. **Take down notes** and solve problems together with the instructor. Keep your calculator handy. Be ready to volunteer to share your solutions to problems given in class.

4. The Online Sync Session will be **recorded** and will be uploaded in our course site.

ASSESSMENTS

1. **Check for Understanding (CfU) assessments** are given in the middle of each module. They are to be answered anytime **online** (within a window of a couple of days) with a time limit. **Coverage of CfUs** includes topics that come before it in the module.

2. **Solutions** to CfUs must be handwritten (or type-written) on an A4 white bond paper, in portrait format, scanned and saved as one file in pdf. Use as filename: Surname – CfU#.pdf.

3. **Late submission** of answer sheets for Check for Understanding will merit 1 point deduction for every 2 min late.

4. **Long Tests** are held onsite. All tests are **closed notes** exams.

5. **Solutions** to tests must be handwritten on an A4 white bond paper, using the **portrait format**.

6. Students are expected to **work independently**. Cite references when necessary. During the quiz/test, you are not allowed to communicate with anyone other than your class instructor.

7. Inform the instructor immediately if there are **conflicts with the schedule** of tests. A make-up exam may only be given for valid reason such as sickness or emergencies and may require submission of medical certificate and other documents.
8. Expect corrected quizzes to be returned two weeks after the test. Students have two days to **report corrections** in checking of quiz/exam after the said requirement is returned.
9. Keep a record of your scores. **Grades on Canvas** are not the official grades of the class.

ATTENDANCE

1. "A limited number of absences is permitted. The total number of absences in a class must not exceed 20% of the total number of meetings per academic term" (2021 LS Undergraduate Academic Regulations Section III.2.1). The number of class meetings is the sum of the number of onsite and online Sync Sessions. Hence, the maximum number of allowable absences is 6. Attendance onsite and online will be checked.
2. Students who arrive (onsite or online) within 15 minutes after the scheduled class time are considered late (0.5 cut) while students who arrive 15 min after class time are considered absent.
3. Students are not allowed to leave the room (also Zoom) once the class has started. Online, the student must ask permission in case he/she has to leave before the end of the class. Students will be marked absent if the total time attended on Zoom is less than 60 min.
4. Inform the teacher in the event of an illness due to Covid-19.

Useful Links

1. For general information about our virtual campus: https://ateneobluecloud.ateneo.edu/
2. The One Stop Student Services for LS: https://sites.google.com/ateneo.edu/ls-one or alternatively, you may use bit.ly/LS-One.
3. The LS Primer, Forging Hope on Reimagined Path, Adaptive Teaching and Learning in the Loyola Schools, SY 2021-2022, ateneobluecloud.ateneo.edu/LS.

K. CONSULTATION HOURS

NAME OF FACULTY	EMAIL	DAY/S	TIME
Felix P. Muga II	fmuga@ateneo.edu	Tue and Thu	02:30-04:30 P.M.

Consultation Hours are subject to change to adjust to the availability of the students. Any changes will be announced within one week from the start of classes.

L. ADDITIONAL NOTES

Important Dates (Subject to changes. Updated dates will be posted in the course site)

Date	Activity	Type
Thu, 11 Aug 2022	**Orientation and Introduction to R Programming**	**On-Site**
Mon, 15 Aug 2022	Data Exploration	On-Site
Thu, 18 Aug 2022	Data Wrangling and Improving Programming Skills	Online-Sync
Mon, 22 Aug 2022	Data Modeling, Part 1	On-Site
Thu, 25 Aug 2022	Data Modeling, Part 2	Online-Sync
Mon, 29 Aug 2022	**National Heroes Day**	
Thu, 1 Sept 2022	Review for Long Test 1	Online-Sync
Mon, 5 Sept 2022	**Long Test 1**	**On-Site**
Thu, 8 Sept 2022	Tokenization	Online-Sync
Mon, 12 Sept 2022	Stop Words	On-Site
Thu, 15 Sept 2022	Stemming	Online-Sync
Mon, 19 Sept 2022	Word Embeddings, Part 1	On-Site
Thu, 22 Sept 2022	Word Embeddings, Part 2	Online-Sync
Mon, 26 Sept 2022	Word Embeddings, Part 3	On-Site
Thu, 29 Sept 2022	Review for Long Test 2	Online-Sync
Mon, 3 Oct 2022	**Long Test 2**	On-Site
Thu, 6 Oct 2022	**Midterm Academic Break (Oct 6 – 8, 2022)**	
Mon, 10 Oct 2022	Regression, Part 1	On-Site
Thur, 13 Oct 2022	Regression, Part 2	Online-Sync
Mon, 17 Oct 2022	Regression, Part 3	On-Site
Thu, 20 Oct 2022	Classification, Part 1	Online-Sync
Mon, 24 Oct 2022	Classification, Part 2	On-Site
Thu, 27 Oct 2022	Classification, Part 3	Online-Sync
Mon, 31 Oct 2022	Dense Neural Networks, Part 1	On-Site
Thu, 3 Nov 2022	Review for Long Test 3	Online-Sync
Mon, 7 Nov 2022	**Long Test 3**	On-Site
Thu, 10 Nov 2022	Dense Neural Networks	Online-Sync
Mon, 14 Nov 2022	Long Short-Term Memory Networks	On-Site
Thu, 17 Nov 2022	Convolution Neural Networks	Online-Sync
Mon, 21 Nov 2022	**Project Proposal Presentation and Submission**	On-Site
Thu, 24 Nov 2022	Project Consultation, Part 1	Online-Sync
Mon, 28 Nov 2022	Project Consultation, Part 2	On-Site
Thu, 1 Dec 2022	**Study Days (Dec 1-2, 2022)**	
Mon, 5 Dec 2022	**Finals Week (Dec 5 - 10, 2022)**	
Thu, 8 Dec 2022	**Finals: Project Presentation**	On-Site
Sat, 10 Dec 2022	**Finals: Project Paper Due**	**Online-Sub**

Appendix : Suggested Readings

Text Mining

1. Silge, Julia and David Robinson. Text Mining with R : A Tidy Approach. (physical copy) : O'Reilly Media; 1st edition (July 18, 2017) Free website version (last built on 2021-04-06): https://www.tidytextmining.com/

A. Natural Language Features

A.1. Tokenization

1. Bender, Emily M. 2013. "Linguistic Fundamentals for Natural Language Processing: 100 Essentials from Morphology and Syntax." Synthesis Lectures on Human Language Technologies 6 (3). Morgan & Claypool Publishers: 1–184.
2. Gagolewski, Marek. 2020. R Package Stringi: Character String Processing Facilities. http://www.gagolewski.com/software/stringi/.
3. Hvitfeldt, Emil. 2019a. Hcandersenr: H.c. Andersens Fairy Tales. https://cran.r project.org/web/packages/hcandersenr/index.html
4. Mullen, Lincoln A., Kenneth Benoit, Os Keyes, Dmitry Selivanov, and Jeffrey Arnold. 2018. "Fast, Consistent Tokenization of Natural Language Text." Journal of Open Source Software 3: 655. doi:10.21105/joss.00655.

5. Perry, Patrick O. 2020. Corpus: Text Corpus Analysis. https://cran.r project.org/web/packages/corpus/index.html

A.2 Stopwords

1. Benoit, Kenneth, David Muhr, and Kohei Watanabe. 2021. Stopwords: Multilingual Stopword Lists. https://cran.r-project.org/web/packages/stopwords/index.html
2. Feldman, R., and J. Sanger. 2007. The Text Mining Handbook. Cambridge University Press.
3. Lewis, David D., Yiming Yang, Tony G. Rose, and Fan Li. 2004. "Rcv1: A New Benchmark Collection for Text Categorization Research." Journal of Machine Learning Research 5: 361–397. https://www.jmlr.org/papers/volume5/lewis04a/lewis04a.pdf
4. Lex, Alexander, Nils Gehlenborg, Hendrik Strobelt, Romain Vuillemot, and Hanspeter Pfister. 2014. "UpSet: Visualization of Intersecting Sets."
5. IEEE Transactions on Visualization and Computer Graphics 20 (12): 1983–1992. doi:10.1109/TVCG.2014.2346248.
6. Luhn, H. P. 1960. "Key Word-in-Context Index for Technical Literature (kwic Index)." American Documentation 11 (4): 288–295. doi:10.1002/asi.5090110403.
7. Mohammad, Saif M., and Peter D. Turney. 2013. "Crowdsourcing a Word–Emotion Association Lexicon." Computational Intelligence 29 (3): 436–465. doi:10.1111/j.1467-8640.2012.00460.x.

8. Nothman, Joel, Hanmin Qin, and Roman Yurchak. 2018. "Stop Word Lists in Free Open-Source Software Packages." In Proceedings of Workshop for NLP Open Source Software (NLP-OSS), 7–12. Melbourne, Australia: Association for Computational Linguistics. doi:10.18653/v1/W18-2502.

A.3 Stemming

1. Arnold, Taylor. 2017. "A Tidy Data Model for Natural Language Processing using cleanNLP." The R Journal 9 (2): 248–267. doi:10.32614/RJ-2017-035.

2. Benoit, Kenneth, and Akitaka Matsuo. 2020. Spacyr: Wrapper to the 'spaCy' 'NLP' Library. https://CRAN.R-project.org/package=spacyr
3. Benoit, Kenneth, Kohei Watanabe, Haiyan Wang, Paul Nulty, Adam Obeng, Stefan Müller, and Akitaka Matsuo. 2018. "Quanteda: An r Package for the Quantitative Analysis of Textual Data." Journal of Open Source Software 3 (30): 774. doi:10.21105/joss.00774.
4. Bouchet-Valat, Milan. 2020. SnowballC: Snowball Stemmers Based on the c 'Libstemmer' UTF-8 Library. https://CRAN.R-project.org/package=SnowballC
5. Harman, Donna. 1991. "How Effective Is Suffixing?" Journal of the American Society for Information Science 42 (1): 7–15. doi:10.1002/(SICI)1097-4571(199101)42:1<7::AID-ASI2>3.0.CO;2-P.
6. Hvitfeldt, Emil. 2020a. Textrecipes: Extra 'Recipes' for Text Processing. https://CRAN.R-project.org/package=textrecipes
7. Lovins, Julie B. 1968. "Development of a Stemming Algorithm." Mechanical Translation and Computational Linguistics 11: 22–31.
8. Miller, George A. 1995. "WordNet: A Lexical Database for English." Commun. ACM 38 (11). New York, NY, USA: ACM: 39–41. doi:10.1145/219717.219748.
9. Ooms, Jeroen. 2020b. Hunspell: High-Performance Stemmer, Tokenizer, and Spell Checker. https://CRAN.R-project.org/package=hunspell
10. Porter, Martin F. 2001. "Snowball: A Language for Stemming Algorithms."
11. Schofield, Alexandra, and David Mimno. 2016. "Comparing Apples to Apple: The Effects of Stemmers on Topic Models." Transactions of the Association for Computational Linguistics 4: 287–300. doi:10.1162/tacl_a_00099.
12. Willett, P. 2006. "The Porter Stemming Algorithm: Then and Now." Program: Electronic Library and Information Systems 40 (3). Emerald: 219–223. doi:10.1108/00330330610681295.

A.4. Word Embeddings

1. Bender, Emily M, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? 🦜." In Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 10–623. FAccT '21. New York, NY, USA: Association for Computing Machinery. doi:10.1145/3442188.3445922.
2. Bojanowski, Piotr, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. "Enriching Word Vectors with Subword Information." Transactions of the Association for Computational Linguistics 5: 135–146. doi:10.1162/tacl_a_00051.
3. Hvitfeldt, Emil. 2020b. Textdata: Download and Load Various Text Datasets. https://CRAN.R-project.org/package=textdata
4. Hvitfeldt, Emil. 2020c. Wordsalad: Provide Tools to Extract and Analyze Word Vectors. https://CRAN.R-project.org/package=wordsalad
5. Robinson, David. 2020. Widyr: Widen, Process, Then Re-Tidy Data. https://CRAN.R project.org/package=widyr
6. Selivanov, Dmitriy, Manuel Bickel, and Qing Wang. 2020. Text2vec: Modern Text Mining Framework for r. https://CRAN.R-project.org/package=text2vec
7. Vaughan, Davis. 2021a. Slider: Sliding Window Functions. https://CRAN.R-project.org/package=slider
8. Vaughan, Davis, and Matt Dancho. 2021. Furrr: Apply Mapping Functions in Parallel Using Futures. https://CRAN.R-project.org/package=furrr

B. References on Machine Learning Methods

B.1 Regression

1. Appleby, Austin. 2008. "MurmurHash." https://sites.google.com/site/murmurhash
2. Benoit, Kenneth, and Akitaka Matsuo. 2020. Spacyr: Wrapper to the 'spaCy' 'NLP' Library. https://CRAN.R-project.org/package=spacyr

3. Helleputte, Thibault. 2021. LiblineaR: Linear Predictive Models Based on the LIBLINEAR c/c++ Library. https://CRAN.R-project.org/package=LiblineaR
4. Hvitfeldt, Emil. 2019b. Scotus: Collection of Supreme Court of the United States' Opinions. https://github.com/EmilHvitfeldt/scotus
5. Hvitfeldt, Emil. 2020a. Textrecipes: Extra 'Recipes' for Text Processing. https://CRAN.R-project.org/package=textrecipes
6. Kuhn, Max, and Hadley Wickham. 2021. Recipes: Preprocessing Tools to Create Design Matrices. https://CRAN.R-project.org/package=recipes
7. Silge, Julia, Fanny Chow, Max Kuhn, and Hadley Wickham. 2021. Rsample: General Resampling Infrastructure. https://CRAN.R-project.org/package=rsample
8. Tang, Cheng, Damien Garreau, and Ulrike von Luxburg. 2018. "When Do Random Forests Fail?" In, 2987–2997. NIPS'18. Red Hook, NY, USA: Curran Associates Inc.
9. Van-Tu, Nguyen, and Le Anh-Cuong. 2016. "Improving Question Classification by Feature Extraction and Selection." Indian Journal of Science and Technology 9 (May). doi:10.17485/ijst/2016/v9i17/93160.

10. Vaughan, Davis. 2021b. Workflows: Modeling Workflows. https://CRAN.R project.org/package=workflows

B.2 Classification

1. Bates, Douglas, and Martin Maechler. 2021. Matrix: Sparse and Dense Matrix Classes and Methods. https://CRAN.R-project.org/package=Matrix
2. Breiman, Leo, Jerome Friedman, Charles J Stone, and Richard A Olshen. 1984. Classification and Regression Trees. CRC press.
3. Frank, Eibe, and Remco R. Bouckaert. 2006. "Naive Bayes for Text Classification with Unbalanced Classes." In Knowledge Discovery in Databases: PKDD 2006, edited by Johannes Fürnkranz, Tobias Scheffer, and Myra Spiliopoulou, 503–510. Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/11871637_49.
4. Hvitfeldt, Emil. 2020d. Themis: Extra Recipes Steps for Dealing with Unbalanced Data. https://CRAN.R-project.org/package=themis
5. Kearney, Michael W. 2019. Textfeatures: Extracts Features from Text. https://CRAN.R-project.org/package=textfeatures
6. Kuhn, Max. 2020. Dials: Tools for Creating Tuning Parameter Values. https://CRAN.R-project.org/package=dials
7. Kuhn, Max, and Davis Vaughan. 2021b. Parsnip: A Common API to Modeling and Analysis Functions.

8. Tibshirani, Robert. 1996. "Regression Shrinkage and Selection via the Lasso." Journal of the Royal Statistical Society. Series B (Methodological) 58 (1). [Royal Statistical Society, Wiley]: 267–288. http://www.jstor.org/stable/2346178
9. Vaughan, Davis, and Max Kuhn. 2020. Hardhat: Construct Modeling Packages. https://CRAN.R-project.org/package=hardhat

C. References on Deep Learning Methods

C.1. Dense Neural Networks

1. Guidotti, Riccardo, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. 2018. " A Survey of Methods for Explaining Black Box Models." ACM Comput. Surv. 51 (5). New York, NY, USA: Association for Computing Machinery. doi:10.1145/3236009.
2. Pennington, Jeffrey, Richard Socher, and Christopher Manning. 2014. "GloVe: Global Vectors for Word Representation." In Proceedings of the 2014 Conference on Empirical Methods in Natural Language

Processing (EMNLP), 1532–1543. Doha, Qatar: Association for Computational Linguistics. doi:10.3115/v1/D14-1162.
3. Shwartz-Ziv, Ravid, and Naftali Tishby. 2017. “Opening the Black Box of Deep Neural Networks via Information.” https://arxiv.org/abs/1703.00810
4. Ushey, Kevin, JJ Allaire, and Yuan Tang. 2021. Reticulate: Interface to ’Python’. https://cran.r-project.org/web/packages/reticulate/index.html

C.2 Long-Short Term Memory ‘

1. Elman, Jeffrey L. 1990. “Finding Structure in Time.” Cognitive Science 14 (2): 179–211. doi:10.1207/s15516709cog1402_1.
2. Hochreiter, Sepp, and Jürgen Schmidhuber. 1997. “Long Short-Term Memory.” Neural Comput. 9 (8). Cambridge, MA, USA: MIT Press: 1735–1780. doi:10.1162/neco.1997.9.8.1735.
3. Kingma, Diederik P., and Jimmy Ba. 2017. “Adam: A Method for Stochastic Optimization.” http://arxiv.org/abs/1412.6980
4. Minaee, Shervin, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. 2021. “Deep Learning–Based Text Classification: A Comprehensive Review.” ACM Comput. Surv. 54 (3). New York, NY, USA: Association for Computing Machinery. doi:10.1145/3439726.
5. Srivastava, Nitish, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. http://jmlr.org/papers/v15/srivastava14a.html. “Dropout: A Simple Way to Prevent Neural Networks from Overfitting.” Journal of Machine Learning Research 15 (56): 1929–1958.

C.3 Convolutional Neural Networks

1. Gage, P. 1994. “A New Algorithm for Data Compression.” The C Users Journal Archive 12: 23–38.
2. Kim, Yoon. 2014. “Convolutional Neural Networks for Sentence Classification.” In Proceedings of the 2014 Conference on doi:10.3115/v1/D14-1181. Empirical Methods in Natural Language Processing (EMNLP), 1746–1751. Doha, Qatar: Association for Computational Linguistics.
3. Pedersen, Thomas Lin, and Michaël Benesty. 2021. Lime: Local Interpretable Model-Agnostic Explanations. https://CRAN.R-project.org/package=lime
4. Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. 2016. “"Why Should i Trust You?": Explaining the Predictions of Any Classifier.” In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1135–1144. KDD ’16. New York, NY, USA: Association for Computing Machinery. doi:10.1145/2939672.2939778.
5. Vosoughi, Soroush, Prashanth Vijayaraghavan, and Deb Roy. 2016. “Tweet2Vec: Learning Tweet Embeddings Using Character-Level CNN-LSTM Encoder-Decoder.” In Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, 1041–1044. SIGIR ’16. New York, NY, USA: Association for Computing Machinery. doi:10.1145/2911451.2914762.
6. Zhang, Xiang, Junbo Zhao, and Yann LeCun. 2015. “Character-Level Convolutional Networks for Text Classification.” In Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, 649–657. NIPS’15. Cambridge, MA, USA: MIT Press.