Chapter 10
High Performance Computing for Understanding Natural Language

Marija Stanojevic
Temple University, USA

Jumanah Alshehri
https://orcid.org/0000-0002-0077-7173
Temple University, USA

Zoran Obradovic
Temple University, USA

ABSTRACT

The amount of user-generated text available online is growing at an ever-increasing rate due to tremendous progress in enlarging inexpensive storage capacity, processing capabilities, and the popularity of online outlets and social networks. Learning language representation and solving tasks in an end-to-end manner, without a need for human-expert feature extraction and creation, has made models more accurate and much more complicated in the number of parameters, requiring parallelized and distributed resources high-performance computing or cloud. This chapter gives an overview of state-of-the-art natural language processing problems, algorithms, models, and libraries. Parallelized and distributed ways to solve text understanding, representation, and classification tasks are also discussed. Additionally, the importance of high-performance computing for natural language processing applications is illustrated by showing details of a few specific applications that use pre-training or self-supervised learning on large amounts of data in text understanding.

DOI: 10.4018/978-1-7998-7156-9.ch010

INTRODUCTION

The exponential data explosion requires developing practical tools for efficient and accurate pattern discovery, classification, representation, trend, and anomaly detection in large-scale high dimensional textual data (Szalay & Gray, 2006). For a decade now, IBM has been using high-performance computing (HPC) to analyze text and create intelligent machines. IBM Watson is a supercomputer that famously leveraged language analysis to win a game of Jeopardy (Hemsoth, 2011).

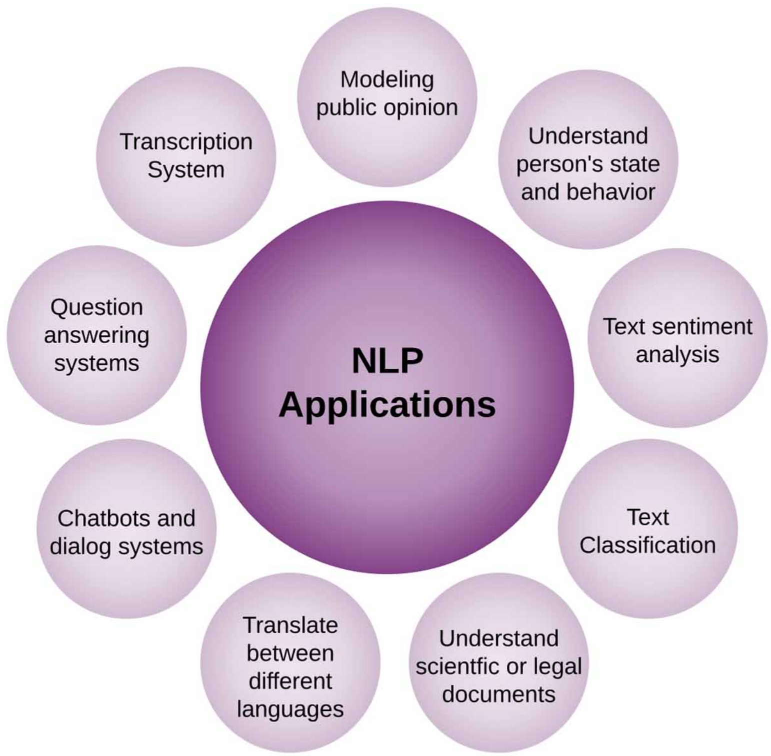
Advances in natural language processing (NLP) are essential for achieving real artificial intelligence. Language is considered one of the most complex human inventions and essential to human intelligence and social integration. Therefore, success in NLP is a prerequisite for fully functioning, artificially intelligent machines.

The industry is currently the largest contributor to NLP development because of its practical importance in handling large amounts of unstructured online data. Understanding public opinion through user-generated text analysis guides more informed decisions, policies, and products. Due to increased use of online social networks, forums, blogs, product reviews, and news comments, it became easy to collect an extensive amount of text needed for understanding opinions and facts about specific topics. Being able to understand those texts fully can shape politics, marketing, and many other fields.

As natural language models have become more complex in recent years, usage of HPC locally or in the cloud has become inevitable in NLP applications. Most novel NLP models are based on neural networks, which forward and backward propagation can be reduced to a vast matrix (tensor) multiplication. Therefore, Graphics Processing Unit (GPU) or Tensors Processing Unit (TPU) hardware is used for faster training. To enhance those models' speed and usability, they are mostly implemented in a distributed manner and expected to run on a high-performance parallel computing system.

Some popular libraries used in implementing and evaluating the most recent natural language models are: NLTK (Loper & Bird, 2002), Gensim (Rehurek & Sojka, 2010), SpaCy (SpaCy, 2020), TensorFlow (Abadi et al., 2016), PyTorch (Paszke et al., 2019), Keras (Chollet, 2017), scikit-learn (Pedregosa, 2011) and all of them support parallel and distributed processing, while most support GPU, and some even run on TPU hardware. Many of those frameworks are easy to learn and have complex neural networks and machine learning modules readily available for use. For those practitioners wanting to create and parallelize their algorithms in python, there is an open-source library, Dask (Dask Development Team, 2019), that natively scales python code. Also, Google has recently developed JAX (Google, 2020), which can transform any python code to allow backpropagation through it. This framework allows an additional training speed up by an innovative combination of operations and simple transformation *pmap,* making the algorithm parallelizable and easy to execute on HPC.

Figure 1. Common NLP Applications



Using those and similar frameworks, people have created data mining and machine learning-based algorithms for different NLP applications. Some of these applications are listed below and summarized in Figure 1.

Some of common NLP applications are:

1. Modeling public opinion from social media and news on different topics (e.g., politics, racism, COVID-19, vaccination);
2. Understand a person's state and behavior (e.g., depression, suicidal thoughts, interest in products, dementia);
3. Sentiment analysis, which goal is to predict the emotion of a given text;
4. Text classification, categorizing text into predefined categories as variables to solve machine learning problems;
5. Understanding and summarizing large amounts of scientific or legal documents;
6. Translation between multiple languages (simultaneously);
7. Chatbots and dialog systems, which can make full-textual conversations with a human agent or another machine;
8. Answering questions automatically, where machines learn how to answer requests coming from humans; and
9. Transcription systems, which aim to teach machines to transcribe voice to text or text to voice.

BACKGROUND

Recent natural language models, such as BERT (Devlin et al., 2018), GPT (Radford et al., 2018), GPT-2 (Radford et al., 2019), ROBERT-a (Liu et al., 2019), ALBERT (Lan et al., 2019), GPT-3 (Brown et al., 2020), and T5 (Raffel C et al., 2020) use transformers (Vaswani et al., 2017) and self-attention mechanisms for text representation learning, using 110 million to 175 billion parameters (weights) to learn from billions of textual examples. Such huge models cannot be handled with any single computer, CPU, or GPU unit, and they are usually optimized and trained in a highly parallel way on a supercomputer.

BERT is among the smaller of the models mentioned above. Its smaller version (BERT base) has 110 million parameters, and the bigger version (BERT large) has 340 million parameters. It takes about 5.4 days to train BERT large on 64 V100 GPUs (Dettmers, 2018). The BERT large model's training takes 34 days on 8 V100 GPUs with full precision and 21 days with half-precision. However, with appropriate parameterization and optimization, NVIDIA successfully trains the BERT large model in only 47 minutes using 1,472 V100 GPUs (Narasimhan, S., 2019).

As very few institutions are equipped with supercomputing power, there are many cloud systems or supercomputers that are offering HPC or supercomputing services for government, academic (e.g., Summit - Oak Ridge National Laboratory, Sierra - Lawrence Livermore National Laboratory, Sunway Taihulight, National Supercomputing Centre), or commercial purposes (e.g., Amazon Web Services - AWS, Google Cloud, Microsoft Azure, IBM Spectrum Computing, Dell EMC HPC). Many commercial solutions offer machine learning as a service on a cloud, which comes with pre-installed software and libraries for machine learning.

PRACTICAL CONSIDERATIONS

Obtaining HPC services to an organization is a crucial decision to make. One must consider many aspects and issues, including privacy, organization utilization, and ways resources would be used. The following are some of the issues involved:

Privacy

- What is the privacy level of that data, and is the user allowed to move the data to external hardware? This question can be problematic for data for which the user has gained unique access. Usually, in such cases, there is a contract that specifies where data can be stored.
- How will HPC hardware store the data, and does the user have options to destroy it entirely? Full data removal should be possible in most of the solutions.

Organization Utilization

- How much do CPU/GPU/storage cost per hour and unit? If the hardware is needed for academic settings, there are many grants and programs through which it can be obtained for free, especially for educational purposes and in smaller amounts. The prices vary between CPU/GPU/TPU units, and they also depend on the amount of RAM given with those processors.

- How much is hard disc space allowed per user? Storage size is rarely the issue in text processing, but some other applications might have this problem.
- How much CPU/GPU computational power exists and can be accessed by one user? This question is one of the significant factors in choosing the right hardware.
- Can the user run multiple processes, and how many of them can be run in parallel? Many of the systems have restrictions on the number of processes that can be run in parallel, and it is essential to understand the level of parallelization.

Resources Usage

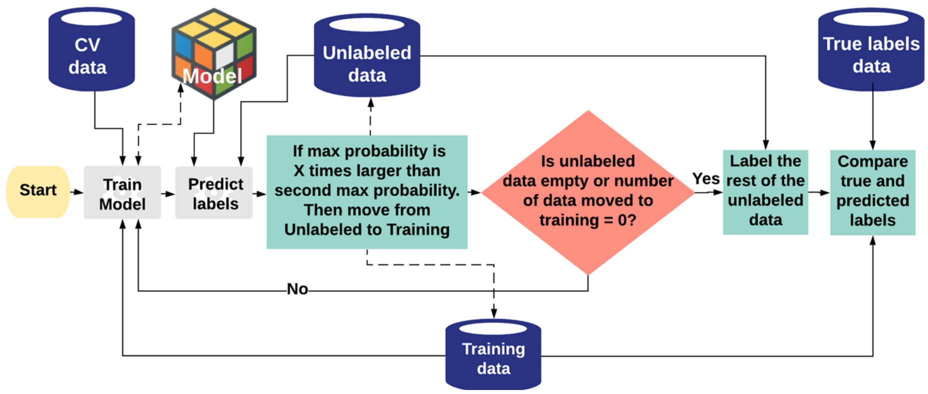
- Is there a wall time constraint? There is a restriction on the maximum duration of a process (wall time). If that exists, the user needs to make sure that progress is saved before the wall time ends and that the program can be continued within a new process from the saved file. It is good practice to save progress more often so that it is not lost in a power outage or other hardware issue.
- Is the algorithm parallelizable? Some machine learning algorithms are not parallelizable, or they may be only partially parallelizable. For example, recurrent neural networks (RNN), which were very popular in NLP before transformers, cannot be fully parallelized because of the hidden layer's serial update. The neuron is waiting for the output of the last neuron in the hidden layer.
- Is the implementation well parallelized? A maximum possible amount of parallel processes should be used to save execution time for big applications. Parallelization would depend on computing, RAM size, and network speed between nodes. It is important to balance them in such a way to get the most out of the hardware. A profiler can be used to understand program resource usage better. Also, in many cases, parallelization is not natural, so additional work needs to be done. In most of the frameworks mentioned above, parallelization will take a few additional code lines, but it may require much more work in some other cases.

APPLICATIONS

One of the significant research and industrial goals is to leverage the dynamics of social media content emerging around news articles (NAs), both at publisher websites (news outlets) worldwide and at social networking services such as Twitter, for intelligence and predictive analytics. Social media and NAs play an essential role in documenting daily societal events (Jin et al., 2017; Ramakrishnan et al., 2014; Rekatsinas et al., 2017; Sakaki et al., 2010; Korolov et al., 2016). For example, in NSF supported project "EAGER: Assessing Influence of News Articles on Emerging Events", the Temple Data Analytics and Biomedical Informatics (DABI) team at Temple University is modeling News Articles and Comments collected from more than 1,000 news outlets worldwide. Transforming the streams of social media and comments at thousands of news outlets (NOs) into data signals is the complicated problem addressed in this project. The researchers then use those signals to foretell the imminence of an (important) event, understand opinions about different topics, and develop sound predictive analytics on top of those signals.

This project requires learning a good text representation of formally (news articles) and informally written texts (comments and social media posts). This is a challenging problem which in the given approach utilizes deep learning models based on building blocks called transformers (Vaswani, A et al., 2017), aimed to discover knowledge from ordered sequences of data. Those models are computationally

Figure 2. Proposed framework for classification of short texts from small amount of labeled data



expensive to train and typically require weeks of distributed GPU processing. On the other side, to model complex spatio-temporal networks of user comments in news outlets and social media containing millions of nodes and links, the team formulate a massive optimization problem that requires parallel processing on CPU nodes with large memory. Additionally, multiple available datasets are used to connect news with other kinds of data to get quantitative and qualitative signals for the underlying machine learning problem. One such application studied at DABI laboratory is crime analysis. Those types of complex networks contain spatio-temporal information and, in general, are extremely large. For example, the DABI laboratory study's crime network consists of more than 4.5 million nodes, even when restricted to US data. Both problems are computationally infeasible without relying on high-performance computing resources.

Domain-Adaptation for Representation Learning of News and User Generated Text

When using social media and news comments text to model public opinion or understand events, researchers and industry have a significant constraint because such texts are short and condensed. Additionally, users-generated texts often contain jargon, sarcasm, links, and emoticons that can change the meaning or the tone of the text. To prevail over those challenges and improve accuracy performance, recent papers proposed algorithms for text classification that require millions of labeled documents (Conneau et al., 2016, Zhang et al., 2015, Yang et al., 2016). A vast amount of data needs to be cumulated and labeled to model public opinion and to ensure representation of different views on the same topic. For example, in recent work (Stanojevic et al., 2019), the authors collected 11.75 million unlabeled tweets on gun advocacy.

It is expensive and very time-consuming to label such vast amounts of text. Most universities and corporations do not have the resources to label such amount of data. Even if those efforts are attempted, it can take years to prepare and characterize enough data. Moreover, when the labeling task is too complicated, or the samples are short and have layered meanings, human labels' accuracy is questionable. In those cases, experts need to be employed for characterizing the example meaning. If machine learning models can help with labeling efforts, researchers and practitioners can focus on modeling and interpreting human behavior and opinions.

In (Stanojevic et al., 2019), a self-supervised framework was developed to label vast amounts of unlabeled data with a few thousand labeled examples (Figure 2). While this approach lowered the amount of required labeled data for up to three orders of magnitude, it also resulted in a small drop in prediction accuracy.

To speed up the training module, the most expensive part of the framework was parallelized to use multiple GPU units on HPC. The most expensive training algorithm used was based on convolutional neural networks (CNN). This allowed for more parallelization than recurrent neural networks, which are more commonly used in text processing. The researchers used nodes with 512 GB of RAM, with two NVIDIA Tesla P100 units, each with 12 GB of RAM. Since GPU units generally have much less RAM, neural network-based training module data was fed in small batches leaving enough RAM space to use more prominent architectures with more parameters. Additionally, incremental training was used so that the time complexity did not increase.

When analyzing an event or opinion, the proposed framework focuses on a specific topic (e.g., politics, economy) in which meaning, sentiment, and distributions of phrases change. For example, the word 'liability' is generally perceived as a negative word. However, economists often use it with a neutral sentiment (Loughran, and McDonald, 2011). The results show that the proposed semi-supervised framework with a training module based on CNN architecture performs the best in predicting millions of tweets labels with just 5000 labeled examples.

Modeling Users Content on Social Media to Understand Public Opinion

News and social media data, while abundant, pose many challenges that limit the potential benefits of machine learning based modeling. Some of the main usage constraints are:

1. The demographic information of users is hidden or not given.
2. Content is short and occasionally incomprehensible without context.
3. Manually labeling millions of posts is challenging for any institution.

The first problem can be solved by using information only from users whose demographic information is publicly available. However, as such a pull of users is tiny; the data may contain bias. As a solution to the second and third challenges, automatic systems need to model text into distinct opinions utilizing users' networks and their published content.

The DABI team explored the utilization of topic-specific news data to fine-tune state-of-the-art models, so they can learn to recognize opinion from social media text (Stanojevic et al., 2019). Specifically, influence of news articles was studied with a different bias on models trained to classify Twitter data. Moreover, performance was evaluated on balanced and unbalanced datasets. The experimental studies revealed that the tuning dataset characteristics, such as bias, diversity of vocabulary, and text style, are determining the success of classification models. On the other side, the data volume was less important. Additionally, it was shown that a state-of-the-art algorithm was not robust on an unbalanced twitter dataset, and it exaggerated when predicting the most frequent label.

To learn better representations of text and reduce training time, pre-trained word embeddings WT103 were used as a starting point. These were created on supercomputers by the training state of the art models with freezing layers on large amounts of English text. ULMFiT architecture is used to learn specific word meaning changes in each domain (social media text on specific topics). Despite using model pre-trained on WT103 as a starting point, the algorithm still required GPU training on HPC. The DABI team used 512 GB RAM node with two NVIDIA Tesla P100 units, each with 12 GB of RAM to learn the word meanings when training with differently biased news data and to classify the twitter data.

Classifying User's Comment Relevancy

Users-generated texts, such as blogs, forums, and online news comments, are a rich public opinion poll. Analyzing such data is essential for social scientists, policymakers, and journalists. Many survey-based studies tried to understand users' behavior by characterizing and categorizing comments in online news (Mishne et al., 2006; Ruiz et al., 2011; Weber et al., 2014; Ziegele et al., 2013).

To better reflect the news and comments semantic relation, a categorization was proposed to label comment-article agreement with one of the four categories: relevant, shared entity and category, same category, or irrelevant (Alshehri et al., 2020). Fleiss Kappa statistics (Fleiss, 1971) showed "fair agreement" of native English speakers in categorizing this alignment. This score confirms that comment relevancy labeling is a challenging task.

In this ongoing research, the DABI team proposed using novel powerful deep learning transformer-based models to understand the level of relevance between articles and comments while working with a limited amount of labeled data. A standard word-level embedding model (Doc2Vec) (Mikolov et al., 2013), recurrent neural model language model (Siamese LSTM) (Mueller et al., 2016), and finally, a pre-trained, transformer language model (BERT) (Devlin et al., 2018) were compared. HPC GPU units were used to train and fine-tune BERT on this task, and it achieved up to 26% improvement in accuracy compared to the previous state-of-the-art model based on LSTM (Mullick et al., 2019). These results confirmed the hypothesis that an architecture based on BERT could capture a deeper level of semantic relatedness between comments and news articles.

CONCLUSION

In conclusion, with the rapid advancements in many NLP models, the use of High-Performance Computers resources became a must. Here is a list of some famous applications of HPC in NLP:

- IBM Watson, which used a supercomputer to solve the question-answering problem (Hemsoth, 2011);
- GPT3, the state-of-the-art language model with 175 billion parameters that deceived humans in many cases, trained on a supercomputer (Brown et al., 2020);
- Google translate, which uses big recurrent neural models trained on GPUs (Wu et al., 2016);
- Grammarly, which leverages transformer-based architectures such that training is parallelized on multiple GPUs, for correcting grammar errors (Alikaniotis & Raheja, 2020).
- Facebook, which uses deep learning, trained on HPC to translate and generate its posts in different languages (Facebook Research, 2016).

Those interested in other academic combinations of NLP and HPC can find other applications in content created by Indiana University[1] and University Santiago de Compostela[2].

ACKNOWLEDGMENT

This research was supported in part by the National Science Foundation grant number IIS-1842183; XSEDE grant number IRI20004; National Science Foundation grant number 1625061; and US Army Research Laboratory grant number W911NF-16-2-0189.

REFERENCES

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., . . . Kudlur, M. (2016). Tensorflow: A System for Large-Scale Machine Learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)* (pp. 265-283). USENIX.

Alikaniotis, D., & Raheja, V. (2020, May 14). Under the Hood at Grammarly: Leveraging Transformer Language Models for Grammatical Error Correction. *Grammarly Engineering Blog*. https://www.grammarly.com/blog/engineering/under-the-hood-at-grammarly-leveraging-transformer-language-models-for-grammatical-error-correction/

Alshehri, J., Stanojevic, M., Dragut, E., & Obradovic, Z. (2020). (Manuscript submitted for publication). Aligning User Comments to the Content of a News Article. *Work (Reading, Mass.)*.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., . . . Agarwal, S. (2020). *Language Models Are Few-Shot Learners.* arXiv Preprint arXiv:2005.14165

Chollet, F. (2017, May 4). Keras-team/keras 2.0.0. *GitHub.* https://github.com/keras-team/keras

Conneau, A., Schwenk, H., Barrault, L., & Lecun, Y. (2017, April). Very Deep Convolutional Networks for Text Classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics:* Volume 1, *Long Papers* (pp. 1107-1116). 10.18653/v1/E17-1104

Dask Development Team. (2019, June 25). Dask 2.0.0: Library for Dynamic Task Scheduling. *Dask*. https://dask.org

Dettmers, T. (2020, September 20). TPUs vs GPUs for Transformers (BERT). *Tim Dettmers*. https://timdettmers.com/2018/10/17/tpus-vs-gpus-for-transformers-bert/

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.

Facebook Research. (2020, May 20). Breaking Down Language Barriers. *Natural Language Processing & Speech*. https://research.fb.com/category/natural-language-processing-and-speech/

Fleiss, J. L. (1971). Measuring Nominal Scale Agreement Among Many Raters. *Psychological Bulletin*, *76*(5), 378–382. doi:10.1037/h0031619

Google. (2020, July 26). Google/jax 0.1.52. *GitHub.* https://github.com/google/jax

Hemsoth, N. (2014, April 19). Bringing Natural Language Processing Home. *HPCwire.* https://www.hpcwire.com/2011/06/09/bringing_natural_language_processing_home/

Howard, J., & Ruder, S. (2018, July). Universal Language Model Fine-tuning for Text Classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (*Volume 1*: Long Papers)* (pp. 328-339). 10.18653/v1/P18-1031

Jin, F., Wang, W., Chakraborty, P., Self, N., Chen, F., & Ramakrishnan, N. (2017, July). Tracking Multiple Social Media for Stock Market Event Prediction. In *Industrial Conference on Data Mining* (pp. 16-30). Springer. 10.1007/978-3-319-62701-4_2

Korolov, R., Lu, D., Wang, J., Zhou, G., Bonial, C., Voss, C., ... Ji, H. (2016, August). On Predicting Social Unrest Using Social Media. In *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)* (pp. 89-95). 10.1109/ASONAM.2016.7752218

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019, September). ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In *International Conference on Learning Representations*.

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.

Loper, E., & Bird, S. (2002). *NLTK: The Natural Language Toolkit.* arXiv Preprint cs/0205028.

Loughran, T., & McDonald, B. (2011). When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks. *The Journal of Finance*, *66*(1), 35–65. doi:10.1111/j.1540-6261.2010.01625.x

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed Representations of Words and Phrases and Their Compositionality. In Advances in Neural Information Processing Systems (pp. 3111-3119). Academic Press.

Mishne, G., & Glance, N. (2006, May). Leave a Reply: An Analysis of Weblog Comments. *Third Annual Workshop on the Weblogging Ecosystem*.

Mueller, J., & Thyagarajan, A. (2016, February). Siamese Recurrent Architectures for Learning Sentence Similarity. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence* (pp. 2786-2792). AAAI.

Mullick, A., Ghosh, S., Dutt, R., Ghosh, A., & Chakraborty, A. (2019, April). Public Sphere 2.0: Targeted Commenting in Online News Media. In *European Conference on Information Retrieval* (pp. 180-187). Springer. 10.1007/978-3-030-15719-7_23

Narasimhan, S. (2020, August 26). NVIDIA Clocks World's Fastest BERT Training Time and Largest Transformer Based Model, Paving Path for Advanced Conversational AI. *NVIDIA Developer Blog.* https://developer.nvidia.com/blog/training-bert-with-gpus/

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., . . . Desmaison, A. (2019). Pytorch: An Imperative Style, High-Performance Deep Learning Library. In Advances in Neural Information Processing Systems (pp. 8026-8037). Academic Press.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Vanderplas, J. (2011). Scikit-Learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). *Improving Language Understanding by Generative Pre-Training*. Academic Press.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language Models are Unsupervised Multitask Learners. *OpenAI blog, 1*(8), 9.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... Liu, P. J. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, *21*(140), 1–67.

Ramakrishnan, N., Butler, P., Muthiah, S., Self, N., Khandpur, R., Saraf, P., ... Kuhlman, C. (2014, August). 'Beating the News' with EMBERS: Forecasting Civil Unrest Using Open Source Indicators. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1799-1808). 10.1145/2623330.2623373

Rehurek, R., & Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*.

Rekatsinas, T., Ghosh, S., Mekaru, S. R., Nsoesie, E. O., Brownstein, J. S., Getoor, L., & Ramakrishnan, N. (2017). Forecasting Rare Disease Outbreaks from Open Source Indicators. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, *10*(2), 136–150. doi:10.1002am.11337

Ruiz, C., Domingo, D., Micó, J. L., Díaz-Noci, J., Meso, K., & Masip, P. (2011). Public Sphere 2.0? The Democratic Qualities of Citizen Debates in Online Newspapers. *The International Journal of Press/Politics*, *16*(4), 463–487. doi:10.1177/1940161211415849

Sakaki, T., Okazaki, M., & Matsuo, Y. (2010, April). Earthquake Shakes Twitter Users: Real-Time Event Detection by Social Sensors. In *Proceedings of the 19th International Conference on World Wide Web* (pp. 851-860). 10.1145/1772690.1772777

SpaCy. (2020, May 19). SpaCy 3.0.0 Industrial-Strength Natural Language Processing in Python. *SpaCy*. https://spacy.io/

Stanojevic, M., Alshehri, J., Dragut, E. C., & Obradovic, Z. (2019, July). Biased News Data Influence on Classifying Social Media Posts. In *Proceedings of NewsIR Workshop @ 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 3-8). Academic Press.

Stanojevic, M., Alshehri, J., & Obradovic, Z. (2019, August). Surveying Public Opinion Using Label Prediction on Social Media Data. In *2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)* (pp. 188-195). IEEE. 10.1145/3341161.3342861

Szalay, A., & Gray, J. (2006). Science in an Exponential World. *Nature*, *440*(7083), 413–414. doi:10.1038/440413a PMID:16554783

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., . . . Polosukhin, I. (2017). Attention is All You Need. In Advances in Neural Information Processing Systems (pp. 5998-6008). Academic Press.

Weber, P. (2014). Discussions in the Comments Section: Factors Influencing Participation and Interactivity in Online Newspapers' Reader Comments. *New Media & Society*, *16*(6), 941–957. doi:10.1177/1461444813495165

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., . . . Klingner, J. (2016). *Google's Neural Machine Translation System: Bridging the Gap Between Human and Machine Translation.* arXiv Preprint arXiv:1609.08144.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E. (2016, June). Hierarchical Attention Networks for Document Classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 1480-1489). Academic Press.

Ziegele, M., & Quiring, O. (2013). Conceptualizing Online Discussion Value: A Multidimensional Framework for Analyzing User Comments on Mass-Media Websites. *Annals of the International Communication Association*, *37*(1), 125–153. doi:10.1080/23808985.2013.11679148

ENDNOTES

1 http://hpnlp.org/

2 http://proxectos.citius.usc.es/hpcpln/index.php/en/