WHITE PAPER

Delivering Mission Impact

IT Spending Transparency Realized with AI

AI-enabled automation of IT procurement categorization and understanding using context-aware BERT language model

Sudhakar Nori, Farshad Saberi-Movahed, Bill Kasenchar

SEPTEMBER 2022

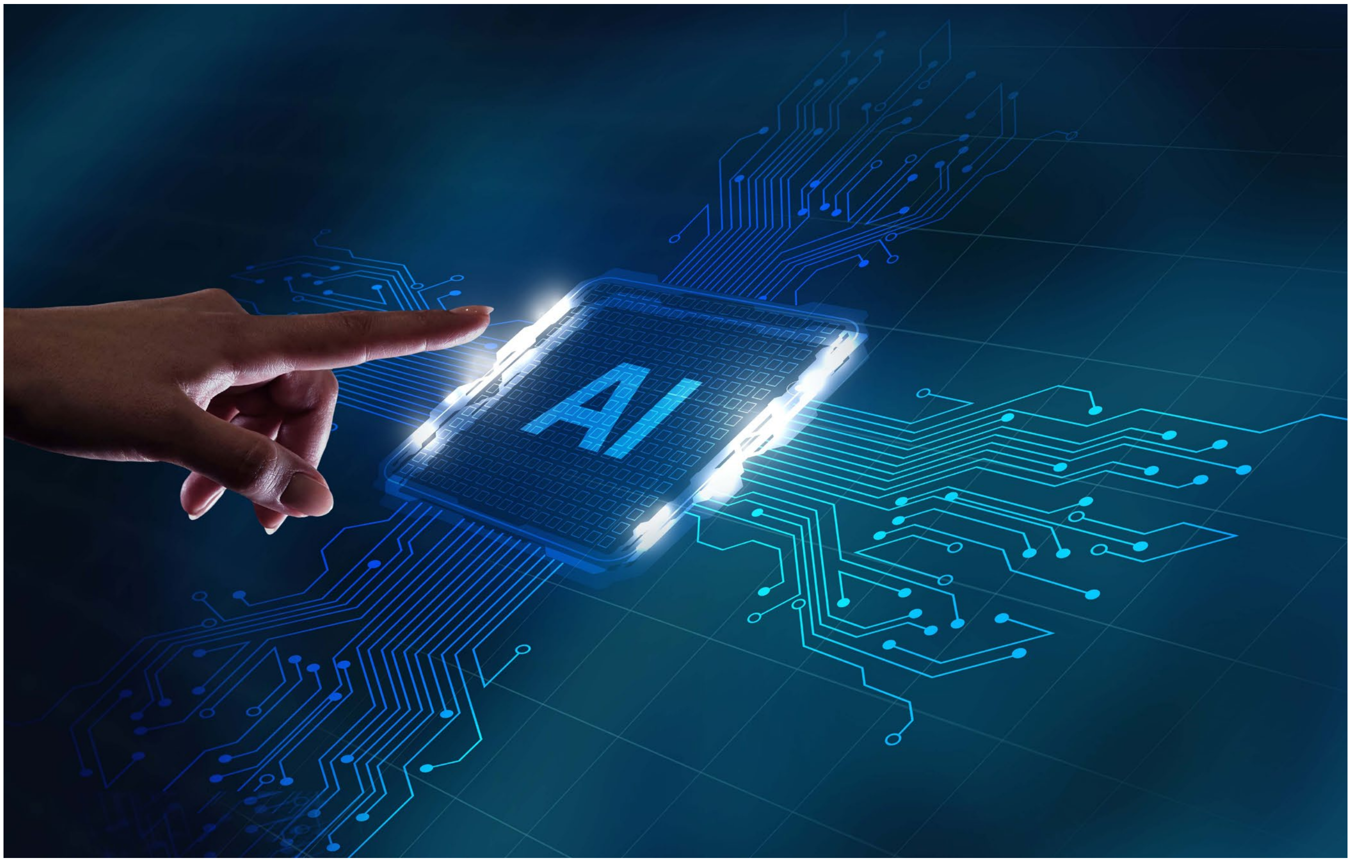
→ REIsystems.com

TABLE OF CONTENTS

Executive Summary

Federal agencies increasingly use Technology Business Management (TBM) best practices to manage and optimize the cost of their IT services and infrastructure. Yet, leveraging the TBM taxonomy to IT procurement data is not always a straightforward process. It requires weeks of manual, error-prone, and low-value work of data analysis and labeling. In this paper, we present our Artificial Intelligence (AI)-based solution to automate the mapping of procurement records into the TBM taxonomy. Our solution is based on a well-known Machine Learning (ML) model called BERT, which we built to capture the complex context of the procurement data. Our model had 97% accuracy on a test dataset for one of our federal clients, and outperformed four other ML models we trained.

In mere minutes, our specialized BERT model can correctly classify millions of dollars' worth of procurement data into Sub-Towers of the TBM Taxonomy. Our approach efficiently and effectively makes IT costs more transparent and offers insight into how and where IT dollars are spent. It also generates higher consistency levels (than human classification) across the federal government, increasing the overall understanding of IT spend.



Introduction

Agencies are leveraging TBM to better understand the cost of IT services, optimize spend, and provide the best value that enables their mission. TBM helps manage IT like a business, supports value conversations, and aligns IT spending with business strategy. By adopting TBM best practices, the federal government can increase transparency, improve service delivery, and empower decision-makers with data for better choices about their technology investments.

Much can be gained by putting technology investments in terms business partners can understand. These practices support federal efforts to promote cost transparency and improve IT management. However, it all starts with understanding the technology the government is purchasing and determining what is IT spend and what is not. This can be done by adopting IT financial management standards, which requires cost reporting by pre-defined groups of underlying technologies, referred to as "Towers" (see Fig. 1). The Towers are part of the TBM taxonomy standard, and associating costs with these functional areas of IT underpin optimization, transparency, and value gain.

TOWERS (v4.0)

Data Center — Compute — Storage — Network — Platform — Output — End User

Application — Delivery — Security & Compliance — IT Management

Fig. 1: IT Towers used in TBM taxonomy

Despite the recognized benefits of TBM, adopting it requires significant analysis of a multitude of data sets to understand which technology is acquired and consumed to support the mission. For most agencies, this involves manually aligning hundreds of thousands of rows of procurement data, an impractical, time-consuming, error-prone, and inconsistent process. Although Product Service Codes (PSC) help categorize costs associated with these procurement data into Tower categories, they are sometimes vague and applied inconsistently. Furthermore, the legacy codes do not align well.

Keyword search has also been used for the categorization task, but it often does not distinguish correctly because the same keyword can mean different things in differing contexts. A top challenge in using this type of Boolean search is knowing which terms to use. Multiple keywords with conflicting mappings can exist in the same record or even as adjacent words. There is some success with building keyword decision trees, but the number of permutations is very large, requiring complex development. In return, the results are questionable and the process requires extensive manual effort for validation and cleanup.

Given the challenges of aligning the procurement data with Tower categories, a mechanism is needed to automatically and accurately review procurement data for consistency. We discuss our approach based on AI and ML. We also present quantitative results from our approach to the procurement data of one of our federal clients.

From a business value perspective, parsing data quickly and consistently can realize economies of scale and consolidation opportunities that can save the government millions of dollars yearly. It can also reduce the barrier of entry for IT cost transparency, resulting in large monetary benefits.

Application of AI for TBM and Procurement Data

CHALLENGES

In recent years, AI and ML have successfully automated many tasks including classification of structured (i.e., tabular data) and unstructured (e.g., text) data into a set of pre-defined categories. However, categorizing procurement records into Towers is challenging, not only for manual analysis but also for AI. On the one hand, millions of rows are generated annually with distinct word combinations, often in incomplete sentences. It becomes more complicated when each Tower has Sub-Towers making 43 distinct categories across the TBM taxonomy (see Fig. 2). On the other hand, the procurement data involve similar terms whose context should be captured before being mapped to the right Sub-Tower.

TOWERS (v4.0)							
Application	Application Development	Application Support & Operations	Business Software				
Compute	Servers	Unix	Midrange	Converged Intrastruction	Mainframe	High Performance Computing	
Data Center	Enterprise Data Center	Other Facilities					
Delivery	IT Service Management	Operations Center	Program, Product & Project Management	Client Management			
End User	Workspace	Mobile Devices	End User Software	Network Printers	Conferencing & AV	IT Help Desk	Deskside Support
IT Management	IT Management & Strategic Planning	Enterprise Architecture	IT Finance	IT Vendor Management			
Network	LAN/WAN	Voice	Transport				
Output	Central Print						
Platform	Database	Middleware	Mainframe Database	Mainframe Middleware	Container Orchestration	Big Data	
Security & Compliance	Security	Compliance	Disaster Recovery				
Storage	Online Storage	Offline Storage	Mainframe Online Storage	Mainframe Offline Storage			

Fig. 2: Sub-towers used in TBM taxonomy.

We expand on the contextual requirements of the procurement data that pose a hurdle for the application of ML models. Synonyms, abbreviations, and Three Letter Acronyms are often part of the data set and must be evaluated consistently in the appropriate context. These units of language are common in procurement data because of the limited field length and the time it takes for data entry.

The federal government and technology alike are plagued with this type of content. These data are usually vague with multifaceted intent from a contract officer's perspective. They are also not entered for the purpose of the downstream IT cost categorization.

Identifying the context is especially difficult for Tower alignment as a single item, vendor, Original Equipment Manufacturer (OEM) or product can appear in two different Towers based on its use. It is difficult for human Subject Matter Experts (SME) to tag these line items consistently, and it wastes those individuals' time. Vendors and OEMs have diversified to the point that it is impossible to distinguish a Tower by vendor/OEM rendering these terms as noise to the system.

To further clarify, we provide some examples below taken from the procurement data.

Example 1: Terms like Srv, Maintenance and Web are commonly classified as Compute, Application Support, and Application Development, respectively. However, the model must look at "Open Text Capture Tool" and categorized these as "Business Software."

> The purchase of an Open Text Capture ToolOption Year 4 Part Number 1000005718-M SEL Capt Srv Vol +Adv Rcg +100K PPY=PA - Maintenance

> The purchase of an Open Text Capture ToolOption Year 4 Part Number 1000005712-M SEL Captiva Web Client =UB - Maintenance

> The purchase of an Open Text Capture ToolOption Year 3 Part Number 1000006644-M SEL Capt Ent Svr+Adv Rcg 1MPPY Bundle=IA - Maintenance

Example 2: This scenario requires a determination between Splunk (security software) and terms like Hard Drive, 12 TB, SATA and Cable. For a correct classification, removing the terms software or license reduces Splunk's relevance so the hardware items (Hard Drive, 12 TB, SATA and Cable) drive the classification to the Sub-Tower of Servers.

> HARD DRIVES FOR SPLUNK SUPERMICRO CABLES AND HARD DRIVE Supermicro Cable CBL-0044L 57.5CM SATA FLAT S-S PBF CABL QTY: 9 EACH @ 1.48 = $ 13.32

> HARD DRIVES FOR SPLUNK SUPERMICRO CABLES AND HARD DRIVE Supermicro 15cm 4-Pin Peripheral Connector to 2 Right Angle SATA Power Extension Cable (CBL-0082L) QTY: 8 EACH @ $2.00 = $16.00

> HARD DRIVES FOR SPLUNK SUPERMICRO CABLES AND HARD DRIVE WD Gold 12TB Enterprise-class Hard Drive SATA 6 Gb/s 7200 RPM 256MB Cache 3.5-Inch Form Factor - 7200rpm 30 EACH @ $350.72 = $10521.04

Example 3: This last example shows how including the vendor's name, such as Oracle, is irrelevant because the company has many different product lines.

- Oracle financials and PeopleSoft are "Business Software"
- Java is an "Application Development" platform
- Oracle database is a "Database"
- WebLogic is "Middleware"

The training of the model must use terms to provide the context instead of "Oracle".

REI'S SOLUTION

To effectively capture the context in the procurement data, we have leveraged an ML model called BERT (Bidirectional Encoder Representations from Transformers) [2]. In the results section, we show how this model surpasses other ML models in categorizing the procurement data into the Sub-Tower. We first overview the BERT model and then we investigate its performance for the mentioned task.

BERT: A VERSATILE AI TOOL TO AUTOMATE TBM TAXONOMY CLASSIFICATION

Historically, computers have had a hard time "understanding" language in its textual form. While these machines can collect, store, and read text inputs very efficiently, they lack basic language context or intent. Fortunately, Natural Language Processing (NLP) and Natural Language Understanding (NLU) can aid with that task. This combined process of linguistics, statistics, ML and AI helps computers not only "understand" human language but decipher and interpret the intent of a specific text. BERT exemplifies these recent advancements in NLP and NLU, which is developed by Google and open sourced to the public. BERT relies on the encoder part of the Transformer model architecture [3], which is also developed by Google. It uses the self-attention mechanism to capture the semantics of words. This mechanism uses elegant yet simple linear algebra operations to establish relationships with different weights between words (or tokens, in the context of BERT). The weights determine the closeness between tokens and capture the context of the sequence.

Oracle E-Business Suite Licenses Oracle licenses. BASE PERIOD   Oracle Financials - Application User Perpetual

Renewal of PeopleSoft Enterprise Licenses and Support for DOJ PeopleSoft Enterprise Human Resources - Enterprise Employee Perpetual, LI-OracleSupport, QTY 76,000

CEC Oracle Java SE Subscription POP: 12 months12 months Oracle Java SE Desktop Subscription for 82,000 named users. Part Number B90201. Electronic Delivery to

Oracle Database Enterprise Edition software and maintenance renewals. Oracle Database Enterprise Edition software and maintenance renewals. Period of Performance: 9/29/18-9/30/19

Oracle Software Update License & Support Renewal Oracle WebLogic Suite

The original BERT model was trained on two self-supervised tasks: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). In MLM, some tokens in the input sequence are randomly masked with a certain probability. The model is then trained to accurately predict the masked tokens (see Fig. 3). In NSP, the task has two sentences (also used for the MLM task) separated by a special token. Then, the model predicts whether these sentences relate to each other. There are some nuances in training BERT, and we have provided resources for the interested reader [4].

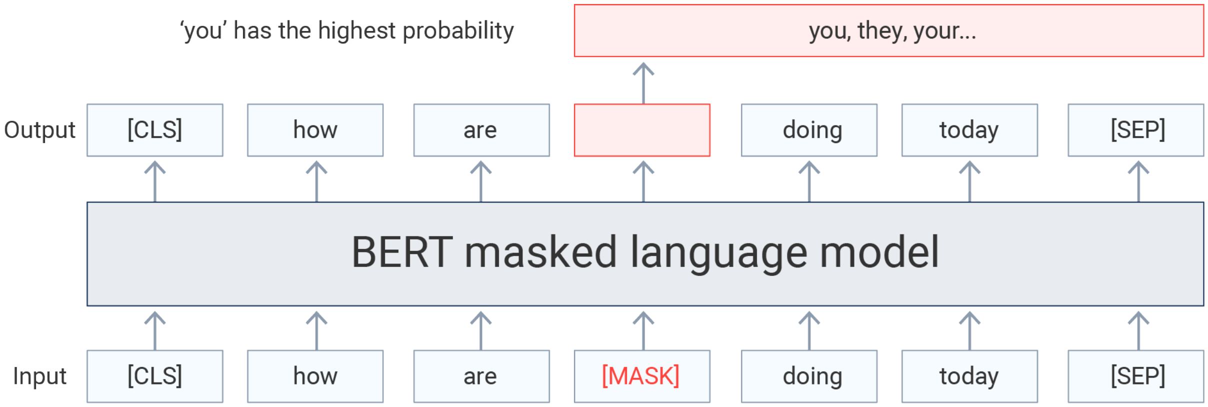


Fig.3: BERT language model pre-training for MLM tasks.

Over the years, different versions of BERT have emerged such as RoBERTa [5], DistillBERT [6] and ALBERT [7]. Each has different model performance, training strategy, and model sizes that range from millions to billions of parameters. These models are usually trained on millions of text records, often crawled from publicly available data sources like Wikipedia, and then presented as pre-trained models in public repositories. These models can distill the contextual information of the input sequences into their last layer's activations and effectively use them for downstream tasks. Increasingly, these pre-trained models are used to classify texts into different categories [8] and for Natural Language Inference (NLI) tasks such as question and answering systems [9]. Unless mentioned otherwise, we refer to BERT as the general family of different models with similar underlying architecture.

Trained on millions of records from the general domain of the English language, the pre-trained BERT model already encodes a deluge of information about the English language. In the context of NLU, it is also referred to as a language model.

To use BERT for tasks beyond MLM and NSP, the model parameters of the pre-trained BERT are used as a starting point in a transfer learning framework for other tasks. Then, additional neural layer(s) are added on top of the pre-trained BERT model to perform the specific task. Subsequently, the whole model including the added layer(s) is trained for much fewer training iterations (aka. epochs). This process is called fine-tuning, which saves hundreds of GPU hours needed to train the original BERT model. The fine-tuning also requires fewer datasets -- hundreds of thousands versus millions. BERT models can be improved with just one extra layer, without needing to make big changes to the model for specific tasks, unlike other models like LSTM. This has made a big impact on how AI is used in language processing and understanding. In fact, this has led to the development of top-performing models that have accuracy close to, or even better than, humans for various NLP/NLU tasks. [10]. For classification tasks, a linear neural layer, or a more complex Multilayer Perceptron (MLP) block, is attached to the Classification (CLS) token representation at the top layer of the pre-trained BERT. The CLS token is used in the pre-trained BERT and is prepended to the beginning of the input sequences while the model is being pre-trained for MLM and NSP tasks. During training, the CLS token attends to other tokens in the sequence through the self-attention mechanism. The final representation of the CLS token contains contextual information of the input sequence [2].

Results

In this section, we present our quantitative results from the application of the BERT model to the categorization of procurement data into IT Sub-Tower categories. We also compare the performance of the BERT model with that of other ML models that we have trained as a benchmark.

FINE-TUNING BERT FOR TBM’S SUB-TOWER CLASSIFICATION

We fine-tuned a pre-trained BERT model to classify expense records into the corresponding Sub-Tower labels for one of our federal government clients. In other words, we specialized the BERT model for procurement data. Because the BERT that we have selected is pre-trained on the general English language domain, while the procurement data consists of IT and financial context.

DATASET PREPARATION

For training purposes, we linked the Award description and the Line-Item description fields in the procurement data of our client. This forms one single field the model runs against. We then split the original dataset of ~275 K expense descriptions into 184,250 and 90,750 records (67%:33% split ratio) to create training and test datasets, respectively. There were 46 Sub-Tower target labels (categories) in the dataset. The way the data was divided for training and testing was done so both sets had the same specific labels. We also applied label encoding to convert the target labels from string format into numeric representation so the BERT model can read them.

TRAINING PROCESS

Because of the proprietary nature of our work, we cannot disclose the full details of the pre-trained BERT model we chose and the intricacies of the fine-tuning process. However, we offer some general information about our model and how we fine-tuned it. To fine-tune the pre-trained BERT model for TBM Sub-Tower classification, we modified its last layer. We then trained the modified BERT model for 10 epochs and ~8 hours. In each epoch, we split the training set into batches to train the model in a manageable way. We then selected accuracy as our metric to evaluate our fine-tuned BERT model on the test dataset. We used Tensorflow v2 [11] deep learning framework for training and evaluation of the model in the Python environment. All operations were performed in one GPU-loaded EC2 instance of AWS.

PREDICTION ACCURACY	
99%	Training set
97%	Test set

Table 1: Accuracy results from our fine-tuned BERT model for TBM Sub-Tower data

BERT PERFORMANCE EVALUATION

We show the training and test accuracies in Table 1. As seen, our model achieved 97% accuracy on the test dataset. Because the difference between the test and training accuracies is insignificant, our fine-tuned BERT model for the Sub-Tower data is not suffering from overfitting. It can thus generalize well to unseen data, which is also confirmed by the results we got from the procurement data of our other clients.

We also experimented to compare how our fine-tuned BERT model performed against other ML models for Sub-Tower classification task. The other models are: one-dimensional Convolutional Neural Network (1D CNN), Random Forest, XGBoost, and k-nearest neighbor (KNN). We also trained these ML models on the same training dataset used for fine-tuning the pre-trained BERT model. The results are in Fig. 4, which shows how superior our fine-tuned BERT model is compared to other ML models in our experiments.

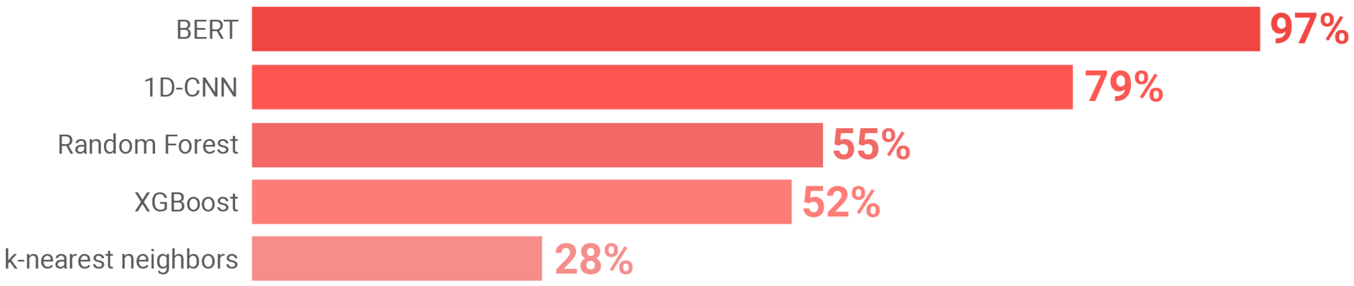


Fig. 4: Comparison of prediction accuracies on test dataset between our fine-tuned BERT model and four other ML models which are 1D-CNN, Random Forest, XGBoost, KNN.

THE BUSINESS IMPACT

The relatively high accuracy of our ML model provides the basis for cost transparency and gives insight into how and where IT dollars are spent. Enhancements and fine-tuning in future revisions are expected to improve the outcome. All in all, our model correctly classified $15B of IT spend data in mere minutes, instead of weeks of manual, low-value, error-prone labeling.

Conclusion

The TBM taxonomy is great for managing the costs and effectiveness of IT services, but it requires significant prep work. To automate this process and make it more accurate and consistent, we developed an AI-based solution that leverages the BERT model. Our value proposition is that we specialized the BERT model, which was pre-trained on general English language context, to the IT and financial context of the procurement data. To achieve this goal, we fine-tuned the pre-trained BERT model on around 185,000 procurement records of one of our federal clients. The fine-tuning was to classify these records to the right TBM Sub-Towers.

The fine-tuned model was 97% accurate on a test dataset (~91,000 records). In other words, our model can correctly categorize 97 records out of 100 records of the IT procurement data into Sub-Towers. We also trained four other ML models: 1D-CNN, Random Forest, XGBoost, and KNN, which had test accuracies of 79%, 55%, 52%, and 28%, respectively. These results show how BERT outperforms against other ML models tested in this study. The reason BERT does better than other models is because it captures the complex context of the procurement data. This data is a great challenge not only to ML models but also to human experts in TBM. Furthermore, the 97% test accuracy of our fine-tuned BERT model indicates our model can generalize well to other unseen procurement data. We verified this by applying our BERT model to the procurement data of our other federal clients. Finally, our solution is not a silver bullet, but a tool in a much bigger toolbox. Going forward, having a better process for tagging to PSC codes at time of procurement, along with training and guidance for contract officers to enter more explicit line-items descriptions for IT spend, will better align IT transparency to mission value.

REFERENCES

1 *Todd Tucker, “Technology Business Management: The Four Value Conversations CIOs Must Have With Their Businesses”, TBM Council, (2016). https://www.tbmcouncil.org/learn-tbm/tbm-book/*

2 *Jacob Devlin, Ming-Wei Chang, Kenton Lee and Kristina Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”, arXiv:1810.04805v2, (2018).*

3 *Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob, Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz, Kaiser and Illia Polosukhin, “Attention is all you need”, In Advances in Neural Information Processing Systems, pp 6000–6010, arXiv:1706.03762v5, (2017).*

4 *Kexin Wang, Nils Reimers and Iryna Gurevych, “TSDAE: Using Transformer-based Sequential Denoising Auto-Encoder for Unsupervised Sentence Embedding Learning”, arXiv:2104.06979v3, (2021).*

5 *Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer and Veselin Stoyanov, “RoBERTa: A Robustly Optimized BERT Pretraining Approach”, arXiv:1907.11692v1, (2019).*

6 *Victor Sanh, Lysandre Debut, Julien Chaumond and Thomas Wolf, ”DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter”, arXiv:1910.01108v4 (2020).*

7 *Zhenzhong Lan and Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma and Radu Soricut, “ALBERT: A Lite BERT for Self-supervised Learning of Language Representations”, arXiv:1909.11942v6, (2019).*

8 *Jason Wei and Kai Zou, “EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks”, arXiv:1901.11196v2, (2019).*

9 *Hoang Van, Zheng Tang and Mihai Surdeanu, “How May I Help You? Using Neural Text Simplification to Improve Downstream NLP Tasks”, arXiv:2109.04604v2, (2021).*

10 *Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy and Samuel R. Bowman, “Glue: A Multi-task Benchmark and Analysis Platform for Natural Language Understanding”, Published as a conference paper at ICLR, https://openreview.net/pdf?id=rJ4km2R5t7, (2019).*

11 *Mart´ın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mane, Rajat Monga, Sherry Moore, Derek Murray, ´ Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viegas, Oriol Vinyals, ´ Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng, “TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems” Preliminary White Paper, November 9, (2015). https://www.tensorflow.org/about/bib#large-scale_machine_learning_on_heterogeneous_distributed_systems*