HARNESSING THE POWER OF NATURAL LANGUAGE PROCESSING: BUILDING INTELLIGENT LANGUAGE MODELS

Dr.K.Gowsic M.E.,Ph.D.,

Associate professor

Department of Computer Science and Engineering

Mahendra Engineering College

T. Arundhathi, S. Monesha, N. Karthick, S. Manoj kumar,

Department of Computer Science and Engineering

Mahendra Engineering College.

ABSTRACT

An HumRRO, we have successfully used natural language processing (NLP) to generate test items for a variety of assessment types. Based on this expertise, we have developed an innovative interface for on-demand automated generation of test items using finely tuned natural language understanding and generation (NLU/NLG) models. The NLU aspect of NLP allows computers to understand the nuance inherent in human speech, which feeds into NLG models that can write natural-sounding language in this case, variable test items that reflects such nuance. Our interface is user-friendly, designed to be understood by item or test developers without prior experience with machine learning or natural language processing. It fine tunes a natural language model on human-written test items, automatically generates new items from this model, and programmatically evaluates the quality of the generated items. When developing content for high-stakes, high-volume testing programs, the circumstances are quite different. Developers must routinely amass and maintain a large bank of items to feed multiple forms that are administered for a finite amount of time before they are replaced with other forms. In addition, the issue of content overlap/redundancy in the item bank becomes more salient as hundreds if not thousands of items must be developed to measure the same set of competencies or knowledge domains. The sheer volume of unique content needed, coupled with development timelines that are typically quite aggressive, necessitates a more strategic development process that focuses on process/operations efficiency, standardization, and waste reduction.

INTRODUCTION

In our experience, natural language generation techniques hold great promise for achieving increased flexibility during item development. The advancements in text generation provided by NLU/NLG‟s ability to comprehend and contextualize words and then predict the most plausible word or words that should follow them are compelling and exciting. Although the application of NLU/NLG to AIG remains in its infancy, its use has surged in other areas, developing a large representation of the English language. As a result, a significant proportion of generated texts can be considered comparatively coherent to human-written text, with little to no additional editing required.

At HumRRO, we have taken advantage of NLU/NLG advances across several assessment formats, including SJTs, personality tests, and interest inventories. We‟ve found that this approach generates diverse, high-quality content across all three item types. Our approach to AIG is agnostic to the specific test content and can theoretically be applied to generate new test content for any testing program. Our experience indicates that NLU/NLG can offer huge advantages over more traditional approaches to AIG:

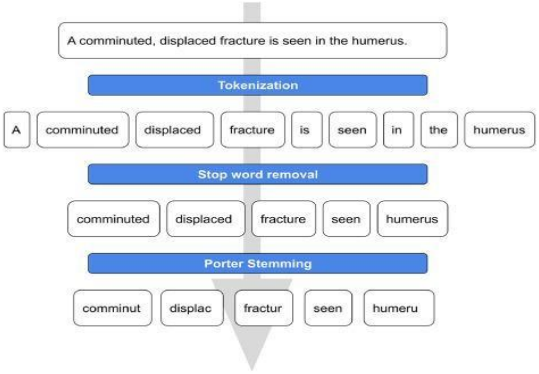

Fig 1: Essential Elements for NLP

Speed or quality in the model development procedure. Underlying these options are parameters that affect the model learning rate. Although the actual speed varies depending on the content input into the program, a quality model could take a few hours longer for the program to develop than a speedy model. A higher quality model is certainly worth the time investment for operational use, but a speedier model may be more reasonable when experimenting with different items, for example. Construct-specific or construct-agnostic item generation. The AIG model can be tuned to understand features that characterize different constructs, such as items measuring different personality traits or items that reflect different proficiency levels. In the model development procedure, users can specify whether to train the model to learn construct labels, and if constructs are specified, users can further specify which constructs they would like to generate items for in the item generation procedure.

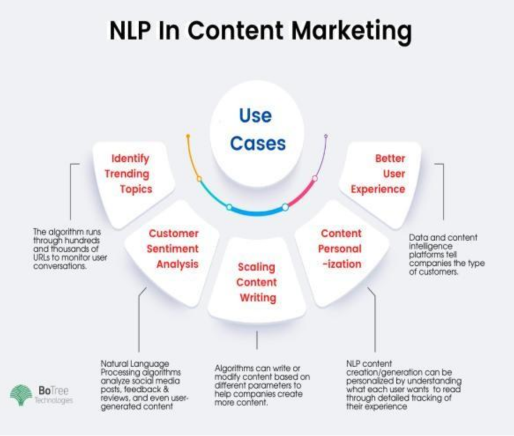
NLP In Content Marketing
Use Cases
Identify Trending Topics
Better User Experience
Customer Sentiment Analysis
Scaling Content Writing
Content Personal -ization
The algorithm runs through hundreds and thousands of URLs to monitor user conversations.
Data and content intelligence platforms tell companies the type of customers.
Natural Language Processing algorithms analyze social media posts, feedback & reviews, and even user-generated content
Algorithms can write or modify content based on different parameters to help companies create more content.
NLP content creation/generation can be personalized by understanding what each user wants to read through detailed tracking of their experience
BoTree Technologies

Fig 2: NLP Processing

To further customize the interface for user preferences, these options can be presented in different formats for different complexity levels. For example, in adjusting speed or quality in model development, one user may prefer simplicity, in which case we provide a simple slider between speed and quality with the underlying parameters adjusted in the background. Another user may prefer greater control, so we also provide a version where the user is able to input values for specific model development parameters. Ultimately, our goal is to make the process and efficiency gains of applying NLU/NLG methods to AIG accessible to everyone. If you are interested in learning more, please contact us for a demonstration. Deterministic or diversified items in the item generation procedure. Underlying these options are parameters that affect the amount of randomness included in the item generation procedure. More randomness means that a greater variety of item content could be obtained, with the tradeoff being higher chance of error such as nonsense items. Text cleaning steps after items are generated can help mitigate some of this error while maintaining diversification. To create engaging and organized articles, we utilize the information extracted from document chunks with respective to blog title, employing custom prompt templates and the powerful OpenAI GPT-3 language model. Our approach involves structuring the article with an introductory section, followed by relevant subheadings that address the chosen blog title. Additionally, we incorporate a section for frequently asked questions (FAQs) and conclude the article with a concise summary.

By leveraging the retrieved data and the capabilities of OpenAI, we generate content for each section and seamlessly merge the resulting responses into a well-crafted article. In conclusion, LangChain represents a significant advancement in the integration of large language models (LLMs) within the field of artificial intelligence. This groundbreaking framework bridges the gap between LLMs and their surrounding environments, allowing for seamless connectivity and enhanced contextual understanding. By leveraging data from various sources and empowering developers to create applications that harness the power of LLMs, LangChain opens up new possibilities for chatbots, question-answering systems, and natural language generation systems.

Through efficient data collection and preparation, the framework optimizes LLMs‟ ability to generate accurate and contextually relevant responses. By leveraging the retrieved document chunks and utilizing custom prompt templates with the OpenAI GPT-3 language model, LangChain facilitates the creation of engaging and well-structured articles. Overall, the collaborative efforts of LangChain and OpenAI revolutionize the integration of AI, unlocking the full potential of language models and paving the way for future advancements in natural language processing and generation.

RELATED WORK

NLP enables computers the ability to "learn" language using data, with some capabilities now approaching human-like performance. Advances in deep learning techniques (e.g., attention mechanisms, memory modules, and architecture search) have made impressive improvements in the NLP landscape (Yogatama et al., 2019). Many NLP tasks, such as question answering, machine translation, reading comprehension, sentiment analysis, and summarization, are often approached using supervised learning on task specific labeled datasets (Radford et al., 2019). However, recent breakthroughs demonstrate that models that are pre-trained on a large unlabeled corpus perform well on many NLP tasks without explicit supervision (Devlin et al., 2018; Radford et al., 2019). These models use a combination of pre-training and supervised fine-tuning where transformers (Vaswani et al., 2017) are used as the backbone of learning. During pre-training, the transformer is trained on a large corpus in an unsupervised fashion such as language modeling (predicting the next word given a context), masked language modeling (predicting a missing word in a sentence from the context), and next sentence predictions (predicting whether two sentences are consecutive sentences). The transformer is then used on various NLP tasks by adding an extra task-specific final layer for fine-tuning. There are at least three factors that affect the performance of transformer-based pre-trained models: (1) size of corpus, (2) availability of computational resources, and (3) expressiveness of model architecture. Because of these factors, the cost and complexity of developing pre-trained models are rising quickly and limit the capability of reproducing high-performance results for those without sufficient resources.

The Vector Institute‟s Recreation of Large Scale Pre-Trained Language Models project (the NLP Project) is an industry-academia collaboration that explores how state-of-the-art natural language processing (NLP) models could be applied in business and industry settings at scale. Developing and employing NLP models in industry has become progressively more challenging as model complexity increases, data sets grow in size, and computational requirements rise. These hurdles limit the accessibility many organizations have to NLP capabilities, putting the significant benefits advanced NLP can provide out of reach. The NLP Project addressed these challenges by familiarizing industry participants with advanced NLP techniques and the workflows for developing new methods that could achieve high performance while using relatively small data sets and widely accessible computing resources. Whereas most NLP research collaborations are designed to produce state-of-the-art models with competitively low error rates, Vector‟s objective was to create a collaborative and scalable learning environment that would allow several companies to gain the hands-on experience necessary to build an end-to-end NLP pipelines and scale their deployment whose primary objective is to produce business value.

NLP enables computers the ability to “learn” language using data, with some capabilities now approaching human-like performance. Advances in deep learning techniques (e.g., attention mechanisms, memory modules, and architecture search) have made impressive improvements in the NLP landscape (Yogatama et al., 2019). Many NLP tasks, such as question answering, machine translation, reading comprehension, sentiment analysis, and summarization, are often approached using supervised learning on task specific labeled datasets (Radford et al., 2019). However, recent breakthroughs demonstrate that models that are pre-trained on a large unlabeled corpus perform well on many NLP tasks without explicit supervision (Devlin et al., 2018; Radford et al., 2019). These models use a combination of pre-training and supervised fine-tuning where transformers (Vaswani et al., 2017) are used as the backbone of learning. During pre-training, the transformer is trained on a large corpus in an unsupervised fashion such as language modeling (predicting the next word given a context), masked language modeling (predicting a missing word in a sentence from the context), and next sentence predictions (predicting whether two sentences are consecutive sentences). The transformer is then used on various NLP tasks by adding an extra task-specific final layer for fine-tuning. There are at least three factors that affect the performance of transformer-based pre-trained models: (1) size of corpus, (2) availability of computational resources, and (3) expressiveness of model architecture. Because of these factors, the cost and complexity of developing pre-trained models are rising quickly and limit the capability of reproducing high-performance results for those without sufficient resources.

As such, the project involved 60 participants: 23 Vector researchers and staff with expertise in machine learning and NLP along with 37 industry technical professionals from 16 Vector sponsor companies. The participants established 11 working groups, each of which developed and performed experiments relevant to existing industry needs. Taken together, through the NLP Project, industry participants benefited by gaining experience with pre-training of large scale language models, attending expert lectures leading to effective knowledge transfer, accessing Vector‟s scientific computing resources, and establishing fruitful collaborations with other sponsor organizations. Notably, insights gained in the NLP Project have informed programs and product development in some participating organizations.

METHODS

This is a joint academic-industrial collaborative project launched in Summer 2019 to explore opportunities as well as promote recent advances in the NLP domain. The project involved 60 participants: 23 Vector researchers and staff with expertise in machine learning and NLP along with 37 industry technical professionals from 16 Vector sponsor companies. The participants established 11 working groups, each of which developed and performed experiments relevant to existing industry needs. The primary objectives of the project were: to foster and widen productive collaboration among academic researchers and industry practitioners on projects in the NLP domain, to help participants in gaining proficiency in building an end-to end NLP pipeline from data ingestion to large scale training and downstream fine-tuning, and to build the capacity for further advances and new lines of businesses in large scale language models in our ecosystem.

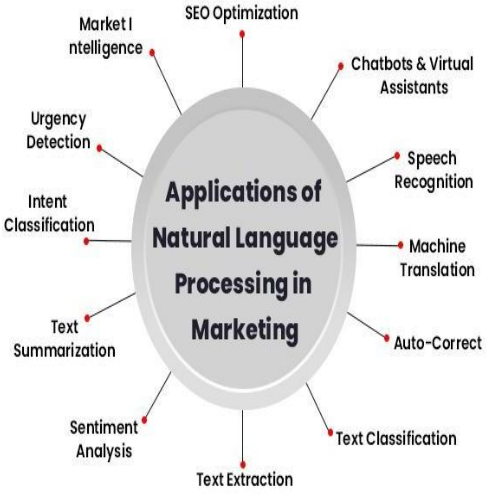

Fig 3: NLP Application

The project was conducted in three phases over 12 months (4 months for each phase); In total 11 working groups were formed to undertake different tasks and activities to achieve the project's main objectives. During the project, weekly meetings were held as ways of communicating current updates and tasks among project members. Commonly found group activities in the weekly meetings were problem solving, decision making, prioritization, and task assignment. Weekly meetings also featured invited guest talks, tutorials on recent advances in NLP and ML from academia and industry, and reading group activities of recent related literature.

During the project, weekly meetings were held as ways of communicating current updates and tasks among project members. Commonly found group activities in the weekly meetings were problem solving, decision making, prioritization, and task assignment. Weekly meetings also featured invited guest talks, tutorials on recent advances in NLP and ML from academia and industry, and reading group activities of recent related literature. Regarding high-performance computing resources, Vector Institute provided participants with 528 GPUs, 6 GPU nodes of 8 x Titan X, and 60 GPU nodes each with 8 x T4, for development and deployment of large-scale transformer-based NLP models. For the model development and evaluation, the implementation from the pytorch transformers library by Hugging Face (2019) was used. 2 Three main project focus areas arose which reflected current industry needs, participants‟ interests and expertise, and opportunities to translate academic advances into real-world application: (1) domain-specific training, (2) pre-training large NLP models, and (3) summarization, question answering, and machine translation. The remainder of this report provides a high-level overview of these focus areas and brief summaries of the working group‟s activities and sub-projects in each area. The participants‟ names are listed in the Appendix section.

The group observed the training loss moving average through 10 epochs of training the BERT model training on a 4 GPU with batch size of 32 and ADAM optimizer. The training loss saturated much faster when starting from the pre-trained weights. Also, when comparing GR, HR, and LR models, the group noted that adding more domain-specific tokens delays the saturation in loss. For all of the downstream tasks except the NER task, the group added a fully connected layer on top of the [CLS] embeddings (or its equivalent in the model). In the NER task, the group used a fully connected layer applied to each token embedding to determine the labels in the experiments. In all of the fine-tuning experiments, the group trained all of the weights (language model + fully connected). For all four tasks, the group split the datasets into training, validation, and test sets in proportions of 80%, 10% and 10% respectively. In each task, the group used early stopping based on the validation loss for all models.

Precision

The precision is the values of properly diagnosis the input data from the heart diseases dataset with total range of heart disease dataset is calculated as

$$\text{Precision} = \frac{\text{Correctly recognized heart diseases}}{\text{Total input dataset of binary value is an lyzed}} \quad (1)$$

Recall

The recall is the values of properly diagnosis of heart diseases of the outcome of heart diseases diagnosis results and it is calculated by

$$\text{recall} = \frac{\text{Accurately analyzed heart diseases}}{\text{Total input dataset of heart disease data} - \text{analyzed}} \quad (2)$$

There are some of the important issues to standing as in feature extraction, classification, and diagnosis of heart disease in the dataset. The optimal accuracy classification techniques must have a high level of recall with accuracy diagnosis results of heart diseases analyzing.

F-Measure

Finally, the F-measure is calculated with precision and recall that can be demoralized in the process of dataset classification and analyzing heart diseases with stages of a prediction method

$$\text{F} - \text{Measure} = \frac{2(\text{precision X recall})}{\text{precision} + \text{recall}} \quad (3)$$

In PR, TM, and SA, pre-training the general BERT language model on the corpus improved the performance results on average 8.5% over the base version. In these three tasks, the highest performance of DistilBERT, RoBERTa, and ALBERT could achieve 95%, 97%, and 95% respectively. These models also seemed to benefit from domain-specific language model customization. For the NER task, the base language models seem to perform better in general. Moreover, by comparing GR and LR versions of BERT and DistilBERT, the group noted that using legal tokens marginally improves the performance compared to using the default general-domain when starting from scratch. However, it still does not beat the impact of using pre-trained weights considering the size of the corpus and the amount of language-model training performed (10 epochs). Even extending the pre-trained model with only some legal tokens degrades the performance for most of the tasks

RESULT ANALYSIS

The pre-training experiment was designed to accommodate some realistic computing resource limitations. In a time-shared computing cluster, GPU resources were allocated on a limited priority basis by Slurm Workload Manager. The working group was able to successfully reproduce the results of BioBERT in part using time-shared computing resources. This work confirms that unsupervised pre-training in general could improve the performance on fine-tuning tasks where large datasets exist. However, the effectiveness of domain-specific pre-training as a way of further improving the performance of supervised downstream tasks may not be wholly substantiated owing to a lack of consistent evidence. Moreover, to facilitate stable training and to accommodate a hardware environment where resources may be reallocated to higher priority users at any time and without warning, the following training features were implemented: (1) storing model weights and gradients at regular time intervals during training; (2) querying and automating job submission from system task scheduler; and (3) automatically restoring training from data chunk and step as GPU resources became available.

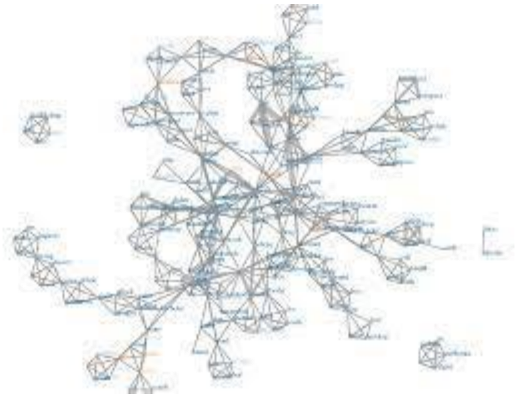

Fig 4: NLP Result Analysis

BERTBASE is the baseline model. The working group started with BERTBASE, then pre-trained it on the PubMed abstracts data (BERTBASE + PubMed). The PubMed corpora used for pre-training consists of paper abstract from millions of samples of biomedical text. While the original BioBERT study considers combined pre-training on PubMed , PMC, and PubMed + PMC together, this working group's model was 3 pre-trained only on PubMed. This was done to check the performance using a smaller amount of data in consideration of the shared computing resources. The PubMed data was processed into a format amenable for pre-training. The raw data consisted of approximately 200 million sentences in 30 Gigabytes. The raw sentence data was batch processed into 111 chunks of ready-to-consume input data for BERT pre-training. Technically, the input for the Next Sentence Prediction (NSP) task was lower-cased sentences with a maximum length of 512 tokens and masked at the sub-token level analogous to the original BERTBASE(uncased). The original BERT models were trained on data with maximum sequence lengths of 128 tokens for the initial 90% of the training and 512 for the remainder. Generally, a longer sequence length is preferable if the corpora tends to have longer passages but this means the amount of data and time to train also increases. In this experiment, the training was maintained at the maximum length of 512 tokens throughout.

For the NER experiment, the working group found that the F1-scores for the vanilla BERTBASE were higher on average than the ones fine-tuned with the in-house BERTBASE+ PubMed. For RE, the group‟s BERTBASE+PubMed outperformed other BERT models, but the differences between models was not significant. Regarding the BioASQ and PubMedQA, the results indicated that pre-training on biomedical domain corpus improves performance on the downstream BioASQ QA task. However, the improvement is not so large as to be entirely convincing and carefully fine-tuned BERT models can perform comparably to BioBERT. Overall, the group‟s exploration confirms that unsupervised pre-training in general could improve the performance on fine-tuning tasks. However, the effectiveness of domain-specific pre-training as a way of further improving the performance of supervised downstream tasks does not significantly outperform the effectiveness of domain-agnostic BERT models considering the high cost of domain-specific pre-training which makes it challenging for most researchers and NLP developers. In the biomedical domain, however, this conclusion may not be wholly substantiated owing to a lack of consistent evidence, particularly in downstream NER and QA tasks.

The working group used a publicly-available corpus of 9,000 U.S. legal agreements as the 9 domain-specific text data. The group investigated the impact of two main factors on training of language models: tokenization and weights initialization. In one experiment, the group used SentencePiece on the legal corpus to generate the same number of cased tokens as BERTBASE(cased)‟s general tokens. The group referred to these domain-specific tokens as „legal tokens‟. Only 36% of tokens are common between legal tokens and general BERT tokens. The group also used a hybrid version of tokenization in which it only added the 500 most frequent tokens in the legal corpus that do not exist as independent tokens in the general BERT. For general and hybrid tokenization (500 legal tokens + BERT tokens) approaches, the group started the training both from the general-domain model weights published with the original papers (i.e., pre-trained initial weights) and from scratch (i.e., random initial weights). Therefore, the group compared six variations of the BERT model:

CONCLUSION

Through the NLP project, launched in the Summer of 2019, the Vector Institute brought together academic and industry sponsors to explore recent advances in NLP. The primary objectives of this project were to foster and widen collaboration between academic researchers and industry applied scientists on several projects, and to build capacity for further advances and new lines of work in large scale language models in our ecosystem. where across tasks and explorations confirmed that unsupervised pre-training or BERT in general could improve the performance on fine-tuning tasks. In some tasks, pre-training the general BERT on a legal corpus improved the results over the performance of the base version and the effectiveness of performing domain-specific pre-training on other tasks using finance data created during the project.

REFERENCES

Araci, D. (2019). FinBERT: Financial Sentiment Analysis with Pre-trained Language Models. In arXiv [cs.CL]. arXiv. http://arxiv.org/abs/1908.10063.

Becker, K. G., Barnes, K. C., Bright, T. J., & Wang, S. A. (2004). The genetic association database. Nature genetics, 36(5), 431-432.

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. arXiv preprint arXiv:1607.06450.

Bhambhoria, R., Feng, L., Sepehr, D., Chen, J., Cowling, C., Kocak, S., & Dolatabadi, E. (2020, November). A Smart System to Generate and Validate Question Answer Pairs for COVID-19 Literature. In Proceedings of the First Workshop on Scholarly Document Processing (pp. 20-30).

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In arXiv [cs.CL]. arXiv. http://arxiv.org/abs/1810.04805. Doğan, R. I., Leaman, R., & Lu, Z. (2014). NCBI disease corpus: a resource for disease name recognition and concept normalization. Journal of biomedical informatics, 47, 1-10.

Drysdale, E., Dolatabadi, E., Chivers, C., Liu, V., Saria, S., Sendak, M., Wiens, J., Brudno, M., Hoyt, A., Mazwi, M., & Others. (2019). Implementing AI in healthcare. https://vectorinstitute.ai/wp-content/uploads/2020/03/implementing-ai-in-healthcare.pdf.

Du, Y., Li, Q., Wang, L., & He, Y. (2020). Biomedical-domain pre-trained language model for extractive summarization. Knowledge-Based Systems, 105964.

Hermann, K. M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., & Blunsom, P. (2015). Teaching machines to read and comprehend. Advances in neural information processing systems, 28, 1693-1701.

Huang, X. S., Perez, F., Ba, J., & Volkovs, M. (2020, November). Improving transformer optimization through better initialization. In International Conference on Machine Learning (pp. 4475-4483).

PMLR. Jin, Q., Dhingra, B., Liu, Z., Cohen, W. W., & Lu, X. (2019). PubMedQA: a dataset for biomedical research question answering. arXiv preprint arXiv:1909.06146.

Khattak, F. K., Jeblee, S., Pou-Prom, C., Abdalla, M., Meaney, C., & Rudzicz, F. (2019). A survey of word embeddings for clinical text. Journal of Biomedical Informatics: X, 4, 100057.

Kim, J. D., Ohta, T., Tsuruoka, Y., Tateisi, Y., & Collier, N. (2004, August). Introduction to the bio-entity recognition task at JNLPBA. In Proceedings of the international joint workshop on natural language processing in biomedicine and its applications (pp. 70-75).

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In arXiv [cs.CL]. arXiv. http://arxiv.org/abs/1909.11942.