Transfer Learning with Transformers: Machine Analysis of the Singapore Parliament Hansard

Joel Ho Eng Kiat[1], **Kingsley Kuan Jun Hao**[1], **Neo Neng Kai Nigel**[1], **Niveditha Nerella**[1], **Noel Mathew Isaac**[1], **Timothy Ong Jing Kai**[1]

[1] CS3244 Project Team 7
National University of Singapore
A0200385N, A0200347U, A0120948A, A0208554A, A0202072Y, A0183548U
{e0407366, e0407328, e0308953, e0445473, e0415881, e0310343}@u.nus.edu

Abstract

We trained and ran Transformer models on the Singapore Parliament Hansard to perform sentiment analysis, name-entity recognition and summarisation, taking advantage of transfer learning via pre-trained models. These were evaluated on a manually labelled dataset and found to give empirically good performance. The models were used to analyse the Hansard of recent years to uncover interesting findings on speakers and entities by sentiment. We release our demo[1] and source code[2].

Introduction

A recent Institute of Policy Studies report (Ng 2021) found that a majority of Singaporean residents are disinterested in politics, leading to political apathy in the population. The report, however, does write that individuals "feel strongly" about certain issues. A follow-up commentary (Teo 2021) describes how their lack of practice and knowledge in political and social issues causes a vicious cycle of further apathy.

One way to get more information is through reading the Singapore Parliament Hansard, which contains verbatim transcripts of the speeches made by politicians in official parliament sessions. Yet understanding the Hansard is not easy, due to the large amount of data provided and lack of more granular categorisation of speeches; even with the gathered text on a particular issue, it is difficult to determine (1) whether the speeches take on a positive or negative view of it, (2) the various entities involved in the discussion, or (3) a summary of the discussion.

In this project, we analysed the Hansard using three different Natural Language Processing (NLP) tasks to match the problems described above: Sentiment Analysis, Name-Entity Recognition (NER), as well Summarisation. Sentiment analysis infers the positive or negative sentiment of portions of speeches, which can be helpful in determining if the speaker is for or against the topic at hand. NER labels persons, organisations, political bodies and other entities in text, which allows for fine grained categorisation of speeches, including references to other politicians in parliament that are not straightforward to search for. Lastly, summarisation produces a succinct overview of the debate which is helpful as bite-sized recaps of parliament proceedings. The use of transfer learning with pre-trained Transformer models allows us to rapidly automate and scale up text processing of the Hansard to democratise information and allow fellow Singaporeans greater access and knowledge of our Parliament sessions.

Transformers

Before 2017, the predominant machine learning approaches for NLP tasks utilised Recurrent Neural Networks (RNNs), specifically Long Short-Term Memory (LSTM) based models (Huang, Xu, and Yu 2015). However, a major weakness of RNNs is the requirement of sequential computation - computing the hidden state of the LSTMs for each input in the sequence requires the hidden state computed from the previous input, hence resulting in a lack of parallelisation. Vaswani et. al. (2017) worked around this problem by proposing a new model architecture, the Transformer, which eschews RNNs in favour of self-attention mechanisms, which learn to focus their "attention" on relevant portions of the input sequence in parallel. This results in a considerable speedup in computation, encouraging larger model sizes and large scale training.

The Transformer architecture consists of multiple stacked self-attention layers as well as fully connected neural network layers. Sentences are first tokenized using byte pair encoding, which splits words according to a vocabulary of the most frequently occurring subwords. After tokenization, tokens are projected onto a fixed dimensional embedding space using an embedding layer, whose weights are learned during training, and used as input to the model. Since the Transformer architecture is not sequential in nature, additional positional information is required, and computed through the addition of a positional embedding with the word embedding. The Transformer self-attention mechanism, briefly summarised, projects a sequence of internal feature representations into sequences of features representing a query and key-value pair (Q, K, V). Then, the scaled dot-product of Q and K is calculated and used to perform a weighted sum of the sequence of V for each sequence position. This results in a new sequence where each feature is a weighted sum of the original sequence according to the relevance as computed by the attention-mechanism. As Q, K, and V are projected using fully connected neural network

[1] https://tinyurl.com/cs3244mlgroup7demo

[2] https://github.com/nus-cs3244-ml-singapore-7/

layers with learnable weights, this self-attention mechanism is also trained end-to-end.

The ability to parallelise computation due to the Transformer architecture has led to the increase in model parameter count as well as deployment of training procedures at scale, leaving other model architectures like RNNs and LSTMs by the wayside. Transformer models can undergo unsupervised pre-training on large corpus of unlabelled data, which brings about greater accuracy for fine-tuning on multiple downstream NLP tasks, often achieving state-of-the-art results, compared to starting on a clean slate (as discussed in the Transfer Learning section). Devlin et al. (2019) introduce BERT, an encoder-only bidirectional Transformer model pre-trained on two problems: a masked language modelling task consisting of the prediction of masked tokens in a sentence, and a next sentence prediction task where the model predicts if one sentence follows another. Liu et al. (2019) improves on BERT by training on larger batches of data over a longer period of time, removing the next sentence prediction objective, amongst other things, to greatly improve the performance of BERT, re-releasing the model as RoBERTa. Conneau et al. (2019) improves on RoBERTa by generalising it into a multilingual language model XLM-R, which can exploit the general structure of languages. We use XLM-R for our subsequent tasks on sentiment analysis and Name-Entity Recognition.

For summarisation, the PEGASUS Transformer model (Zhang et al. 2019) was chosen instead. The model was pre-trained using a self-supervised approach. Firstly, sentences in the text data that most closely represented the overall meaning of the text were masked and replaced with a token. Then, the model was trained with the objective of generating the masked sentences. Such an approach was effective in training the model to produce accurate summaries as it closely resembles the task of summarisation. Due to the difference in training objectives, BERT and its descendant models are not as effective here.

Transfer Learning

Transfer learning is widely applied in the Natural Language Processing field (Ruder et al. 2019), due to the need for a varied but accurate text corpus as well as the huge computational power needed to process the data. Additionally, the Transformer models used are complex enough to require large chunks of time and memory to train properly. Transfer learning in this case is highly beneficial as pre-trained models are widely available for use after pre-training optimised by professionals (Liu et al. 2019). This allows for a large head start in deploying our model that builds upon state-of-the-art technology with cutting-edge implementations.

Practical benefits aside, transfer learning also fits in the conceptual understanding of NLP. For all text analysis, there is a need to understand the syntax and semantics of language, regardless of what the resulting output of the analysis needs to be. Hence a model can first be pre-trained to understand words, context and structure to a high enough accuracy, then the pre-trained model can be fine-tuned to suit the task at hand. This is similar to how pre-training gives a broad overview (high-level) of the topic at hand, while fine-tuning provides more specific guidance (low-level) towards the desired task output, based on the assumption that the model's knowledge in the pre-training task is transferred to the specific task (Wang and Zheng 2016). In essence, the inductive bias of the pre-trained model is used to constrain the search space of the final model, such that less training is needed to obtain the right hypothesis. Thus this method of sequential transfer learning, where a model is trained for one task first and then another, is also deemed as an example of inductive transfer learning (Ruder 2019).

Hence, Transformers with transfer learning allows us to fine-tune the model without requiring much data for the desired task. For our project, we use a (comparatively) small dataset of Singapore Parliament Hansard speeches for a second round of fine-tuning, meaning that our Transformer models are first pre-trained for general language understanding, then fine-tuned on the three NLP tasks listed above, then further fine-tuned to suite the Singapore context.

Experimental Setup

Data Preparation

The Hansard is openly accessible to the public through the Singapore Parliament page, where users can search for individual topics ("sections", i.e. Bills/Motions/Oral Question and Answers) or full day sessions ("sessions"). We work on full day Hansard sessions and scrape them from the period September 2012 to March 2021, resulting in roughly a decade worth of data spanning three sessions of Parliament, each with a different set of Members of Parliament (MP). A script was implemented to automatically scrape the sessions as raw JSON files. The records before September 2012 are not used due to a change in formatting of the Hansard, making it more difficult to parse. These JSON files are parsed and formatted to extract the session titles, MP names and speeches of each subsection. Since speeches of MPs with political appointments were labelled with their relevant political title together with their name (e.g. "Minister of Finance (Mr Heng Swee Keat)"), we sanitise the names for consistency across all the sessions. After formatting, the speeches are divided into paragraphs as displayed on the Hansard webpage.

We choose four sections of the Hansard for manual labelling as a gold standard database. For NER, we annotate using the IOB format (Ramshaw and Marcus 1999), with the entity types defined in OntoNotes 5 (Weischedel et al. 2013), using the ner-annotator package in pip. For sentiment analysis we manually label the speeches with positive(1)/negative(0) sentiment. Similarly, training data for summarising is obtained by manually summarising the text.

Surrogate Datasets

As our manually labelled Singapore Hansard datasets are comparatively small, we select larger surrogate datasets that we use to fine-tune models suited for the general tasks of sentiment analysis, NER, and summarisation.

For sentiment analysis, we select the Stanford Sentiment Treebank dataset, a subset of the General Language Understanding Evaluation benchmark dataset, which we call

SST-2 (Wang and Zheng 2016). The dataset is composed of sentences taken from movie reviews and labelled by human annotators with a binary sentiment, positive or negative. While sourced from movie reviews, the sentences are diverse enough to be suitable for general sentiment analysis.

We also select the HanDeSet (Abercrombie and Batista-Navarro 2018) dataset, which is composed of speeches made in the UK parliament from 1997 to 2017 manually labelled with a positive/negative label. Though it is much smaller than GLUE SST-2, it is better aligned to our goal of performing sentiment analysis on parliament speech.

For NER, we select the OntoNotes 5 (Weischedel et al. 2013) dataset, a large corpus of text of various genres in three languages, which was manually labelled with entity tags in IOB format. We select this dataset as it is diverse in both the types of text used, as well as text language, making it suitable for general NER.

For summarisation, we select the Multi-News (Fabbri, Li, and Radev 2011) dataset which consists of news articles and their human-written summaries, written professionally by editors. While there exists other datasets for summarisation that are more closely related to governmental proceedings (Kornilova and Eidelman 2019), these are highly technical and complex involving aspects of legislature, which are not as similar to the debates in the Singapore Hansard where social and political issues are also addressed.

Model Training

We define and train our models using PyTorch, an open source machine learning framework that provides modules for neural network creation as well as automatic differentiation and gradient descent optimisation. We also use HuggingFace Transformers, a library built on top of PyTorch that provides a collection of Transformer model architectures, as well as pre-trained Transformer models that are both pre-trained on large corpus of unlabelled data as well as fine-tuned for specific tasks.

As modern Transformer models contain many model parameters, they necessitate the use of GPUs, which are able to perform parallel computation quickly and efficiently for training and inference. For our experiments, we use a combination of personal machines (NVIDIA GTX 1080Ti GPU), School of Computing Compute Cluster instances (NVIDIA Titan V GPU), as well as Google Colab, a cloud platform that provides free access to GPUs to encourage machine learning research and applications.

Our models are trained using backpropagation and an adaptive stochastic gradient descent optimiser, in particular AdamW (Loshchilov and Hutter 2019). Unlike the classic stochastic gradient descent algorithm, which uses a global learning rate for all parameters, adaptive stochastic gradient descent algorithms compute individual adaptive learning rates for each model parameter. These are derived using estimates of the first and second moments of the gradient. This allows the optimiser to find the ideal learning rate for each parameter, and benefits scenarios with sparse gradients in deep neural networks or where there is large amounts of data and model parameters.

Due to the addition of new output layers with randomly initialised weights for each new task, learning rate warm-up was used in order to stabilise training, where the learning rate was linearly increased from 0 to the target starting learning rate over 1 epoch. The learning rate was then linearly annealed from the starting learning rate to 0 at the end of training.

Sentiment Analysis For sentiment analysis, we fine-tune the pre-trained XLM-RoBERTa Base model as originally presented by Conneau et al. (2019). Note that we do not use the Large model as the model does not fit on our available GPUs. We experiment with multiple stages of fine-tuning on both our surrogate datasets as well as our manually labelled Singapore Hansard dataset. In order to create different subsets to perform training and validation on, we randomly split our manually labeled Singapore Hansard data 80/20, where 80% is used as a training set and 20% is used as a validation set. However, as there is a class imbalance in our dataset, with around twice as many positive instances than negative instances, we also perform oversampling of the negative instances in the training set in order to balance the dataset and prevent the model from being biased towards either positive or negative. HanDeSet is similarly split 80/20 and is already roughly balanced, while SST-2 has well defined training and validation sets.

For all datasets, we train on the training set for 10 epochs, evaluating the accuracy, F1 score, precision, and recall of the model on the validation set after each epoch. We perform early stopping when the performance on the validation set no longer improves or starts to decrease, signalling that the model is starting to overfit on the data. Hyperparameter search for the learning rate was also performed in {7e-6, 1e-5, 3e-5}, with the best performing model on the validation set taken as our final model.

NER For NER, we use both the XLM-RoBERTa Base model as originally presented, as well as a XLM-RoBERTa Base model that has already been fine-tuned on OntoNotes 5, as provided by (Ushio and Camacho-Collados 2021). We then fine-tune both models on our labelled dataset, again split 80/20 where 80% is used as the training set and 20% is used as the validation set. We train for 30 epochs on the training set, evaluating the F1 score, precision, and recall of the model on the validation set after each epoch. We again perform early stopping in order to prevent overfitting. Hyperparameter search for the learning rate was performed in {7e-6, 1e-5, 3e-5}, with the best performing model on the validation set taken as our final model.

Summarisation For summarisation, we use the PEGASUS model (Zhang et al. 2019) that was pre-trained on news datasets. The model is then fine-tuned with the Singapore Hansard dataset. As mentioned earlier, the Singapore Hansard dataset was manually summarised, and as such the dataset size is much smaller, so it was only trained for 5 epochs. The model was evaluated by its ROUGE-1 and ROUGE-2 scores (Lin 2004). ROUGE computes the similarity of two texts by measuring their n-gram overlap, analogous to recall. As there seemed to be no significant improvement in the summaries on manual inspection, we group the pre-trained and fine-tuned model together in our discussion.

Results and Discussion

Sentiment Analysis

We evaluate the performance of our trained models for sentiment analysis on the validation set of our labelled Singapore Hansard dataset and present results in Table 1. The model trained only on SST-2 showed acceptable performance when evaluated on our dataset, highlighting its good generalisation performance even on data from a different domain. On the other hand, the models that were further fine-tuned on HanDeSeT did not perform well. This is surprising as the HanDeSeT dataset is conceptually similar to our Singapore Hansard dataset. Upon closer inspection of the data, however, the language used in the UK parliament shows stark differences when compared to that used in Singapore parliament. In particular, the language used is archaic, often dramatic, and much more adversarial. We hence theorise that the model trained on HanDeSeT does not generalise well to other domains, even other parliamentary speech.

We observe similar results on models that were then further fine-tuned on the training set of our Singapore Hansard dataset. Models that were initially fine-tuned on HanDeSeT showed decreased performance, even when compared to the model that was only fine-tuned on Singapore Hansard. On the other hand, the model that was initially fine-tuned on SST-2 then further fine-tuned on Singapore Hansard was the best performing model. This fits with our conceptual understanding of transfer learning, where the search space of the problem is gradually narrowed from more general to more specific, and the use of pre-trained models help to constrain the search space such that less data and training is required.

NER

We evaluate the performance of our trained models for NER on the validation set of our labelled Singapore Hansard dataset and present results in Table 2. We use (Nakayama 2018) to calculate the metrics on entity level matches in order to match the approach used in other works. While the model pre-trained on OntoNotes 5 did not show good performance by itself, we observe that it does improve the performance of our model when later fine-tuned on our training data. This again aligns with our conceptual understanding and demonstrates the effectiveness of transfer learning.

Interpretation of Results

Sentiment Analysis and NER

With good performance empirically demonstrated for the above tasks, this shows that our models may be utilised for analysing the Singapore Hansard. After running the model on our database of Hansard sessions, what kind of juicy information can we extract?

Much of parliament is about consensus building and passing of laws, but it is the controversial, argumentative portions of parliament that gather headlines. In a similar vein, we will like to know which speakers have been more critical during Parliament sessions and see what insights we can gleam from there. This can be done by looking at the speakers with the lowest sentiment scores from our data, as a low sentiment would arise from a larger proportion of statements that are classified as having negative opinions or intent.

The speaker with the 3rd lowest score is, as expected, an opposition MP, Mr Gerald Giam, with a score of 0.420. As part of the opposition, it is natural that he challenge government policy, questioning decisions and putting forth alternative proposals; the first two tasks here will unsurprisingly lead to a lower sentiment score. Out of all opposition members, what could explain his low sentiment score? Some digging points to his personal website (geraldgiam.sg), where he publishes his Parliamentary Questions as spoken in Parliament, showcasing a persona as a harsh (but fair) critic of the government. This is indeed true, as he describes himself as a “policy wonk”, living out his passion for “critiquing and analysing” government policy (Chew 2021).

The speaker with the 2nd lowest score of 0.418 is, surprisingly, Mr K Shanmugam, Minister of Law and Home Affairs. As a veteran MP from the establishment party, one might expect his rule to be less combative in nature; however, it is important to include his portfolio to better explain his low sentiment score. Since many articles of law are addressed and debated in Parliament, it is expected that Minister Shanmugam will speak on behalf of his articles often enough in Parliament. With opposition members critiquing policy, it is unsurprising that Minister Shanmugam has to defend his policy and offer rebuttals on the opposition’s ideas.

Even more surprisingly, the speaker with the lowest score of 0.414 is Mr Lim Biow Chuan, a backbencher PAP MP representing Mountbatten. It is not immediately obvious why his sentiment score is so low, but looking at the top NER entities in his speeches may elucidate certain details. In fact, one organisation that stands out is the mention of “CASE”, or the Consumer Association of Singapore, which has Mr Lim as its President. With the standard Hansard search, it is not possible to directly search for “CASE” due to the inclusion of other texts which also refer to case (e.g. police cases), hence by using NER, we are able to elicit Mr Lim’s pet topic of CASE as mentioned in his parliament speeches. This is reinforced through CASE’s aims, which directly advocates for the “pushing of legislation”, as it has “relentlessly lobbied to the government” as stated on its website (CASE).

We also analyse the most negatively classified entities. When considering recurring entities (more than 50 instances), we observe that the entities with the lowest sentiment correspond to controversial topics of the day, including: ”Internal Security Act” (related to recent news on the radicalisation of two youths), ”Ms Liyani” (on the inequality of law enforcement and proceedings for foreign workers) and ”KPMG” (on their audit on the Aljunied-Hougang Town Council). We may also search for sessions where these entities were mentioned most, allowing us to explore the full context and debate surrounding these topics.

Summarisation

While the ROUGE scores of the pretrained and fine-tuned models were very similar, both models would output a summary that was irrelevant to the Singapore context, bringing

Model Name	Learning Rate	Early Stopping (Epoch)	Accuracy	F1 Score	Precision	Recall
base-sst-2	7e-6	5	0.780	0.834	0.820	0.849
base-handeset	3e-5	6	0.447	0.547	0.587	0.512
base-sst-2-handeset	3e-5	10	0.561	0.691	0.637	0.756
base-sh-sentiment	7e-6	10	0.856	0.889	0.894	**0.884**
base-sst-2-sh-sentiment	1e-5	2	**0.879**	**0.904**	**0.938**	0.872
base-handeset-sh-sentiment	3e-5	10	0.773	0.828	0.818	0.837
base-sst-2-handeset-sh-sentiment	3e-5	2	0.841	0.873	0.911	0.837

Table 1: Results for Sentiment Analysis task on different models. Note that the model name indicates which datasets the model was fine-tuned on, where sh-sentiment is our manually labelled Singapore Handsard sentiment dataset.

Model Name	Learning Rate	Early Stopping (Epoch)	F1 Score	Precision	Recall
asahi417/tner-xlm-roberta-base-ontonotes5	-	-	0.343	0.274	0.458
xlm-roberta-base-sh-ner	3e-5	25	0.786	0.742	0.837
xlm-roberta-base-ontonotes5-sh-ner	1e-5	24	**0.819**	**0.778**	**0.864**

Table 2: Results on Name-Entity Recognition task for different models. Note that the model name indicates which datasets the model was fine-tuned on, where sh-ner is our manually labelled Singapore Handsard NER dataset.

in external vocabulary that on manual inspection made no sense. For example, the “Governance of Aljunied-Hougang Town Council” motion moved by Mr Heng Swee Keat produced the following summary with the fine-tuned model:

> A town council leader in Singapore has been stripped of her position after a judge found she ”acted dishonestly and in breach of their fiduciary duties” and ”lacked integrity and candour,” the New York Times reports.

Although the motion did call for the recusal of Mr Low Thia Kang and Ms Sylvia Lim over financial matters of the town council, it is not accurate to say that they were “stripped” from their roles. The model also reports on only one person, which again shows how the model does not capture the full story. Importantly, Ms Sylvia highlighted the importance of due process during the motion, where the full justice process should be taken before committing to any action. In fact, many of the rebuttals offered by the opposition parliaments did not make it into the summary. Finally, there was no mention of the New York Times in the document; such fabrications or erroneous text can lead to misrepresentations of parliament sessions. It is clear that this model is unsuitable for use in the summarisation of parliamentary proceedings, at least not in the context of the Singapore Hansard.

Conclusion

We have explored the numerous advantages of both Transformer models as well as transfer learning, and run experiments with pre-trained models fine-tuned on both surrogate as well as our own manually labelled dataset. We empirically demonstrate good performance for sentiment analysis and NER when applied to the Singapore Hansard. However, a similar approach applied to summarisation does not result in an improvement, and that it generalises poorly to the Singapore Hansard.

We propose that this is due to starting from a model that was pre-trained on data sourced from the western context and is thus not adapted to the context and structure of Singaporean parliamentary proceedings. Furthermore, it is too labour intensive to manually summarise large amounts of data from the Singapore Hansard, resulting in sparse training data. Future research can be done in improving the results of summarisation through either exploring alternative machine learning approaches that are less biased, or through large-scale annotation of the Singapore Hansard dataset.

The results of NER and sentiment analysis obtained can be used to perform exploratory data analysis to find interesting patterns in the data. Some of the potential applications of this data in the Singapore context include

- Developing a parliamentary search engine using the entities extracted to help users sift through Hansard data based on the entity types, entity frequency, sentiment range, etc.
- Aggregating all extracted data into a summarised report which helps members of the public better understand issues discussed in parliamentary sessions and improve their political knowledge.

Roles and Reflections

Nigel scraped the Hansard and cleaned the data. Nigel and Niveditha worked on the manually labelled dataset. Kingsley trained and evaluated the models for sentiment analysis and NER, as well as ran the final models on all the scrapped data. Timothy and Joel ran the model for summarisation. Joel additionally trained the model for summarisation. Noel deployed the model online and summarised some findings. Nigel interpreted the results for the report.

Joel - I learnt about the details of how NLP models are implemented and trained, and how to quantitatively evaluate summarisation models using ROUGE. I also learnt the challenges behind training these models, and how encoder-decoder models work. While I had experience in PyTorch, this was another challenge on its own.

Kingsley - While I have experience in deep learning research and PyTorch, I had not explored Transformers in depth before. Utilising pre-trained Transformer models from HuggingFace was both painless and rewarding as experiments could be done quickly. However, converting datasets to be compatible with the models was difficult and often required workarounds.

Nigel - I learnt about the technical implementation details of NLP and the HuggingFace library. I felt the pain in manually labelling the Hansard. I appreciated how my interest for current affairs could be utilised in this cross-disciplinary project on computational social science.

Niveditha - I had some theoretical knowledge, but no technical experience. Through this project I learned skills from labelling, formatting data, to learning about the Spacy vs hugging-face transformers for NLP implementation. I was involved in model research, working on the dataset itself, and analysing results which fell in-line with my major.

Noel - I gained a better understanding of NLP using transformers and learnt that adapting pre-trained models using transfer learning allows one to achieve state-of-the-art performance with relatively lower training time and data. I also learnt more about using PyTorch and HuggingFace for model training and inference.

Timothy - I learnt how neural networks can be applied to achieve state-of-the-art performance in various machine learning tasks. I also learnt how to implement machine learning libraries in Python, namely PyTorch and the huggingface transformers library, to solve real world problems in NLP.

References

Abercrombie, G.; and Batista-Navarro, R. 2018. HanDeSeT: Hansard Debates with Sentiment Tags. doi:10.17632/xsvp45cbt4.2. URL https://data.mendeley.com/datasets/xsvp45cbt4/2.

CASE. 2021. CASE - About Us. URL https://www.case.org.sg/aboutus.aspx.

Chew, H. M. 2021. It's like 'night and day': Workers' Party MP Gerald Giam on being an MP versus an NCMP. URL https://www.channelnewsasia.com/news/singapore/workers-party-mp-gerald-giam-interview-ncmp-aljunied-grc-14398172.

Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv* ISSN 23318422. doi:10.18653/v1/2020.acl-main.747.

Devlin, J.; Chang, M. W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference* 1(Mlm): 4171–4186.

Fabbri, A. R.; Li, I.; and Radev, D. R. 2011. Multi-News: a Large-Scale Multi-Document Summarization Dataset and Abstractive Hierarchical Model .

Huang, Z.; Xu, W.; and Yu, K. 2015. Bidirectional LSTM-CRF Models for Sequence Tagging URL http://arxiv.org/abs/1508.01991.

Kornilova, A.; and Eidelman, V. 2019. BillSum: A corpus for automatic summarization of US legislation. *arXiv* (2017). ISSN 23318422. doi:10.18653/v1/d19-5406.

Lin, C.-Y. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, 74–81. Barcelona, Spain: Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W04-1013.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv* (1). ISSN 23318422.

Loshchilov, I.; and Hutter, F. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Nakayama, H. 2018. seqeval: A Python framework for sequence labeling evaluation. URL https://github.com/chakki-works/seqeval. Software available from https://github.com/chakki-works/seqeval.

Ng, J. S. 2021. Most S'poreans aren't interested in politics, but feel strongly about some policy issues: IPS study. URL https://www.todayonline.com/singapore/most-sporeans-arent-interested-politics-feel-strongly-about-policy-issues-ips-study.

Ramshaw, L. A.; and Marcus, M. P. 1999. Text Chunking Using Transformation-Based Learning 157–176. doi:10.1007/978-94-017-2390-9_10.

Ruder, S. 2019. *Neural Transfer Learning for Natural Language Processing*. Ph.D. thesis, National University of Ireland, Galway. URL https://ruder.io/thesis/.

Ruder, S.; Peters, M.; Swayamdipta, S.; and Wolf, T. 2019. Transfer learning in natural language processing tutorial. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Tutorial Abstracts* (2010): 15–18. doi:10.18653/v1/N19-5004.

Teo, K. K. 2021. Are Singaporeans really politically apathetic? URL https://www.todayonline.com/commentary/are-singaporeans-really-politically-apathetic.

Ushio, A.; and Camacho-Collados, J. 2021. T-NER: An All-Round Python Library for Transformer-based Named Entity Recognition. In *Proceedings of EACL: System Demonstrations*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* 2017-December(Nips): 5999–6009. ISSN 10495258.

Wang, D.; and Zheng, T. F. 2016. Transfer learning for speech and language processing. *2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA ASC 2015* 1225–1237. doi:10.1109/APSIPA.2015.7415532.

Weischedel, R.; Palmer, M.; Marcus, M.; Hovy, E.; Pradhan, S.; Ramshaw, L.; Xue, N.; Taylor, A.; Kaufman, J.; Franchini, M.; El-Bachouti, M.; Belvin, R.; and Houston, A. 2013. OntoNotes Release 5.0. doi:10.35111/xmhb-2b84. URL https://catalog.ldc.upenn.edu/LDC2013T19.

Zhang, J.; Zhao, Y.; Saleh, M.; and Liu, P. J. 2019. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. *arXiv* ISSN 23318422.