Semantically Conditioned Language Models for Political Text Generation

Gökberk Özsoy
ETH Zurich
goezsoy@student.ethz.ch

ABSTRACT

Conditional text generation is an active research field within Natural Language Processing domain. It has many practical applications such as story generation, dialogue systems, or synthetic text generation for data augmentation [7]. In this paper, we develop a 3-step fully automatized and scalable pipeline for generating high quality synthetic text corpus with semantically conditioned language models. Applying the model on a political speech corpus leaves a performant classifier in high confusion for discriminating real and synthetic texts, proving the ability of our pipeline.

1 INTRODUCTION

Text generation refers to producing consistent, and natural texts in a human language. In recent years, transformer-based large-scale language models have been excelled on generating fluent and diverse sentences. However, without explicit guidance, the generation can head to any direction due to stochastic decoding. Conditional language modeling deals with this problem, enabling us to generate semantically conditioned samples.

In this paper, we propose a 3-step approach to generate a high quality synthetic text corpus, reflecting the topics, and ideas present in the original corpus. It is self-contained, fully automatized, and requires no labels. First, we choose GPT-2 as our language model, and fine-tune it slightly using real texts from the original corpus. This allows GPT-2 to specialize in domain specific term usage during generation. Second, we extract the ideas embedded in real texts of the original corpus as keywords via a state-of-the-art keyword extractor. We will use these keywords as our semantic constraints during generation. Third, we employ Keyword2Text [9] weighted decoding algorithm, that works on top of GPT-2 for conditional language generation. It simply shifts the output probabilities of GPT-2 for each token towards input keywords (hence the intended topic), where the shift magnitude is measured by word vector similarities between keywords and tokens in GPT-2 space. Finally, we train real vs. fake BERT classifier using real samples from validation set, and newly generated fake samples, and save real predicted fake samples to our high quality synthetic corpus.

We evaluated our model on U.S. Congressional Record Corpus, containing 395.983 high quality political speeches. After running the above pipeline several times with different K2T hyperparameters, we gathered 3400 high quality fake speeches. Sampling the same amount real texts from validation set, and training an additional real vs fake classifier shows us that the accuracy is 0.68 on average, given 0.5 is random guessing. This proves our method's validity, and it can be easily applied to any domain.

2 RELATED WORK

Conditional language modeling is an activate area of research. Current efforts can be classified into 3 main categories [13]: Fine-tuning, re-training, or post-processing. Fine-tuning branch aims to fine-tune all or part of pre-trained language model's parameters to produce conditioned text. For example, Prefix Tuning [8] freezes language model's parameters, and learns task specific continuous prefixes through the backpropagation. The learned prefixes are then used as a prompt which can steer the generation. Re-training branch aims to train the language model from scratch which can accept the condition as input. CTRL [6] labels the whole dataset with control codes, and prepends these to beginning of the sentence during training. For generation, inputting one of these control codes as prompt is enough to generate conditioned text. Post-processing branch is the most effortless, and environment friendly one, where the aim is freezing the language model and only changing its output distribution towards desired feature for each token. One of our pipeline's part, Keyword2Text [9] belongs to this family, and it requires no additional training. Furthermore, FUDGE [12] trains an attribute predictor which changes the probability distribution along the way according to satisfied attribute probability so far.

Human evaluation is still the gold standard for checking quality of generated texts. However, it is an expensive and long procedure, that is not readily available. BERT is shown to be effective on single sentence classification tasks, such as SST-2, and CoLA [3]. Related with our task, FakeBERT [5] uses a combination of convolutions and BERT to reach 98.90% accuracy on test set. Furthermore, Sentiment Preserving Fake Online News Generation [1] generates fake reviews using a two stage approach. It feeds the GPT-2 with real review to

Code is available at https://github.com/gozsoy/conditioned_speech_gen sample a fake one, and only saves it if the BERT classifies both as having the same sentiments. Human evaluators measure the fluency of the generations. Our method is self-sufficient, requiring no labeled data but a corpus, and it filters the generated texts in terms of fluency, grammar, and coherency by employing a choosy real vs. fake BERT classifier.

3 DATASET

The dataset for this project is U.S. Congressional Record Corpus, for the years 1995-2020, with the speaker metadata [4]. Metadata contains speaker name, age, gender, party, state, term type (either Senate or House), speech itself, and speech date. In total, the corpus consists of 1.085.838 single speeches, addressed by 1472 different speakers.

At this point, we analyzed the raw speeches to be sure about their quality, and decided to process the corpus further following these steps: First, we eliminate duplicate speeches, and remove newline character. Second, we remove speeches that do not start with ‘.’ which are plain law texts of the amendments proposed by the corresponding speakers and read to audience during the meeting. Third, we compute the word count per speech for the remaining corpus, and investigated the outliers, namely the longest and shortest texts. We found that the ones that belong to above 99.5th percentile are still plain law texts, and removed them. In addition, we observed that our model tends to generate short sentences with common words. It turns out that speeches which belong to below 30th percentile in terms of word count dominates the data, and they are mostly about greeting, yielding speech to other congressmen, or declaring her vote for the ongoing amendment. We removed these as well. Fourth, for the remaining pure law texts, we sensed one common phrase, which is ’as follows:’ said by the clerk when she is about to read the proposal, and removed speeches including this phrase.

After these steps, we believe that the corpus only contains speeches which have high political opinion and emotion. In Table 1, we present corpus size after each processing step mentioned above. We split the resulting corpus randomly with fixed seed into 95:5 train to test ratio.

Processing Step	Corpus Size
Raw Corpus	1.085.838
Step 1: Remove duplicates	937.860
Step 2: Remove not starting with ’.’	658.043
Step 3: Remove longest and shortest	425.098
Step 4: Remove including ’as follows:’	395.983

Table 1: Effect of preprocessing steps on corpus size. Size measured as single speech count.

4 METHOD

In this section, we explore steps of the political text generation pipeline. First two sections describe the fundamental concepts behind the generator. Third section explains the remaining parts of the generation pipeline, and how they interact with each other. Finally, we extensively report the implementation details. Pipeline as a whole is visualized in Figure 1.

4.1 Language Modeling

A language model is a probability distribution over words. Given a sentence y of length N, the language model p assigns a probability to this sentence $P(y) = P(y_1, y_2, ..., y_N)$. This joint probability distribution can be factorized as:

$$P(y) = \prod_{i=1}^{N} p(y_i|y_{<i}) \tag{1}$$

Today, language models are often parameterized by transformer based architectures such as GPT-2. These models have hundreds of millions of parameters and are trained on very large scale corpora, which enables them to capture semantic and linguistic knowledge successfully.

4.1.1 Fine-tuning a Language Model. A pre-trained language model is required to be further fine-tuned to perform task-specific generation. Fine-tuning will adjust the parameters of the language model, so that sentences from corresponding task will be receive higher probabilities. In our task, we also fine-tune the GPT-2 model with the speeches from the training set. Here, the aim is not to memorize the training set, but to slightly shift GPT-2 into political domain.

4.2 Keyword2Text [9]

The task of text generation is decoding sentences from natural language with the help of language model p. During decoding, the language model p produces a categorical distribution for each word in the sentence $p(y_i|y_{<i})$. We can manipulate this distribution for semantically conditioning the generation towards the given topic or idea, which is done by Keyword2Text (K2T). K2T together with the fine-tuned GPT-2 are the backbones of our realistic political speech generator. Visualized as middle box in Figure 1.

K2T is a model agnostic, weighted decoding algorithm used for controlled text generation. Its idea is simple and intuitive: given a keyword or topic, it adds a shift to the probability distribution over vocabulary towards semantically similar words. The shift is calculated based on cosine similarities between Glove word embeddings of keyword and words in vocabulary:

$$p'(y_i|y_{<i}) = logp(y_i|y_{<i}) + \lambda \cdot max(0, cos-sim(\gamma(y_i), \gamma(w))) \tag{2}$$

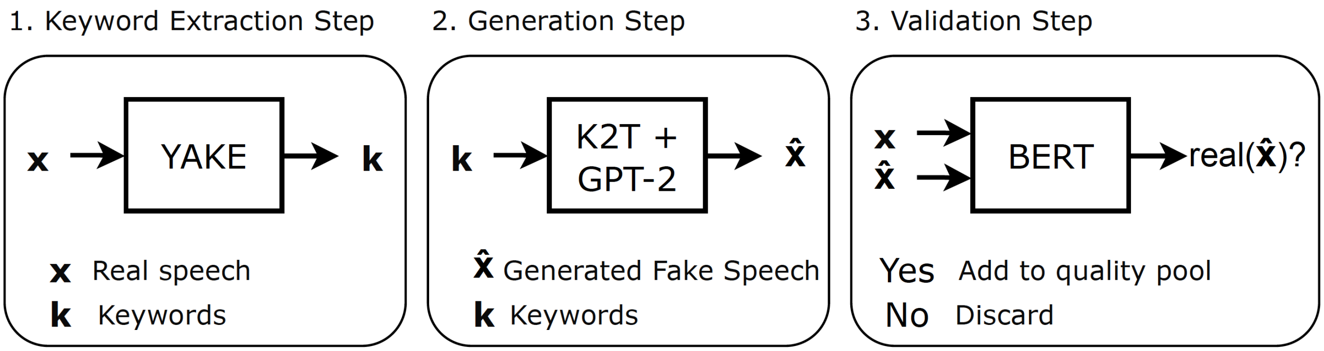

Figure 1: Pipeline for generating high quality synthetic political speech corpus. GPT-2 in middle box is fine-tuned. Figure style influenced from [1].

where w is given keyword or topic, γ is Glove word embedding vector, λ is the strength of the shift towards w. Log probability is used for numerical stability.

Given this adjusted probability distribution, any decoding algorithm can be employed to generate text such as beam search or nucleus sampling. Although, this is enough for generating texts infused with the given topic, it cannot guarantee that the given topic, or keywords appear in the text. Thus, the authors exponentially increase λ according to below formula for higher semantic shift as the length of generated sentence increases.

$$\lambda_i = \begin{cases} \lambda_0 \cdot \exp \frac{100(i-i_n)}{N-|W_i|-i_n} & \text{if } i < N - |W_i| \\ \infty & \text{otherwise} \end{cases} \tag{3}$$

where λ_0 is the initial shift strength, N is sentence length, i is the index of the current word y_i which is about to be sampled, i_n is the index of previous guide word appeared during generation, $|W_i|$ is the total number of remaining guide words to be appeared in the future steps. When the remaining space is only enough for remaining keywords, they deterministically put them as in the second condition by making their probabilities to ∞.

K2T requires no additional training, and can generate fluent and coherent text with given idea embedded all around. We will use it on top of our fine-tuned GPT-2 to generate political texts on specific topics such as environment, gun control, healthcare, education, and tribute to community leaders.

4.3 Generation Pipeline

Fine-tuned GPT-2 with K2T is the most critical part of the overall pipeline. However, for automatized and scalable fake political speech generation, we need to support it with a keyword extractor, and quality validator.

4.3.1 YAKE Keyword Extractor [2]. YAKE is an unsupervised keyword extraction algorithm which relies on statistical features extracted from given text to select the most important keywords. Each candidate word is described with statistical features such as casing, term position, and term frequency normalization and ranked from most to least important. YAKE does not need to be trained on a particular set of documents, neither it depends on dictionaries, external-corpus, size of the text, language or domain. Experiments show that it significantly outperforms unsupervised and supervised approaches.

We will use YAKE as our political keyword extractor given the real political speech from the validation set. This step eliminates the necessity of reading many speeches one-by-one to extract keywords, and clears the way of large scale fake text generation. Extracted keywords will be used as inputs to our K2T + GPT-2 generator. Visualized as left box in Figure 1.

4.3.2 BERT [3] as Real vs. Fake Classifier. Decoding can produce ungrammatical, incoherent, or unnatural text, because at the end, it is still relying on bare sampling from a conditional distribution. Human evaluator can easily detect the mistakes, and correct them to increase the fluency of the generated text, but this is not scalable.

We employ BERT classifier, which is already proven to be successful for text classification tasks as the quality control mechanism of the pipeline. The aim is to train the it for discriminating the real and fake speeches for each generation cohort. Real speeches are labeled as 1, and are the ones from the validation set we used for keyword extraction. Fake speeches are generated using those keywords and labeled as 0. After training the model with 0.8 of the dataset, we evaluate our model on the remaining 0.2, and save the false positives to quality pool, which only hosts the highest

Experiment									Performance				
ID	Model	Prompt Length	Token Count	λ_0	Top-p	Det BS	Mode	Gnte	TPR ↑	F1-score ↓	Accuracy ↓	AUROC ↓	AUPRC ↓
1	Plain GPT-2	2	256	5.0	0.9	×	max	×	0.009	0.995	0.993	0.999	1.0
2	Plain GPT-2	2	256	5.0	0.9	×	max	✓	0.008	0.992	0.992	1.0	1.0
3	Finetuned GPT-2	2	256	5.0	0.9	×	max	×	0.082	0.915	0.915	0.976	0.974
4	Finetuned GPT-2	2	256	5.0	0.9	×	max	✓	0.121	0.918	0.912	0.97	0.967
5	Finetuned GPT-2	2	256	5.0	0.9	✓	max	×	0.008	0.991	0.987	1.0	1.0
6	Finetuned GPT-2	2	256	20.0	0.7	×	max	×	0.014	0.99	0.99	0.999	0.998
7	Finetuned GPT-2	2	256	10.0	0.7	×	random	✓	0.08	0.937	0.917	0.978	0.988
8	Finetuned GPT-2	15	256	10.0	0.7	×	max	✓	0.062	0.903	0.897	0.97	0.975
9	Finetuned GPT-2	15	256	5.0	0.9	×	max	✓	0.143	0.888	0.876	0.945	0.944
10	Finetuned GPT-2	15	512	5.0	0.9	×	max	×	0.14	0.906	0.894	0.954	0.955
11	Finetuned GPT-2	15	512	5.0	0.9	×	max	✓	0.126	0.877	0.874	0.95	0.947
12	Finetuned GPT-2	25	256	5.0	0.9	×	max	✓	**0.146**	**0.882**	**0.874**	**0.939**	**0.931**

Table 2: Performance evaluated on 5 different metrics for different fine-tuning regimes and K2T hyperparameters. See Evaluation for detailed explanation of these variables. Bold indicates best performance.

quality ones. We will evaluate the validity of this claim in evaluation section.

4.4 Implementation Details

We used pretrained GPT-2 Medium (355M parameters) from the Huggingface library, and employed gradient checkpointing, gradient accumulation, and mixed precision to reduce utilized GPU memory. We truncated speeches above 256 tokens, and fine-tuned the model on training set with 64 effective batch size, 2e-5 learning rate for the maximum of 3 epochs. We also experimented with truncation of speeches above 512 tokens, where we halved the actual batch size to 4, while effective batch size remained the same.

For K2T, we slightly modified their code to accept fine-tuned GPT-2 checkpoint. K2T is a slow generation algorithm, where single generation can last from 6 to 40 seconds depending on the experiment settings. To increase the throughput, we divided the extracted keywords into 10 shards, for each of which, we start separate GPU jobs parallely.

For keyword extraction, we used the code from YAKE's github page. Default implementation outputs 10 keywords per text, and we randomly sampled 3 of them without considering their importance. Also, we only used unigrams.

Lastly, we used pretrained DistilBERT Classifier (66M parameters) from the Huggingface library. For each fake and real sample, we truncated above 100 words to prevent the model to deduct output from length. 100 words sentence is still long enough to thrive in terms of political opinions. We fine-tuned the model for 10 epochs with 1e-5 learning rate without any hyperparameter tuning. The train to test ratio is 75:25, and we trained 4 independent models for each of the 4 folds for capturing and saving all high quality generations.

5 EVALUATION

In this section, we mention about performance metrics, experiment settings, and corresponding performances.

5.1 Performance Metrics

We utilized False Positive Rate (FPR), F1-score, Accuracy, Area Under the ROC curve (AUROC), and Area Under the Precision-Recall curve (AUPRC) as performance indicators. We used their scikit-learn implementations. Each experiment is conducted on a balanced real vs fake dataset, where real samples are labeled as 1, and fake ones as 0.

5.2 Experiment Settings

For Table 2, 'Model' means if we used not fine-tuned GPT-2 (plain) or fine-tuned one (Finetuned). 'Prompt length' means how many words we provided the generator as initial prompt. Here '2' indicates that for each validation sample which starts which 'Mr.' or 'Madam', its corresponding fake sample will start with these as well. If validation sample starts differently, we use prompt length of 0. 'Token count' indicates maximum token count during fine-tuning above which is truncated. 'λ_0' is the shift strength of K2T. 'Top-p' is the parameter for

Random State	Performance			
	F1-score ↓	Accuracy ↓	AUROC ↓	AUPRC ↓
42	0.63	0.69	0.75	0.77
0	0.64	0.69	0.76	0.78
12	0.66	0.69	0.76	0.77
41	0.66	0.69	0.76	0.77

Table 3: Testing the high quality synthetic political speech dataset against different cohorts of real speeches sampled from validation set using different random states

nucleus sampling. 'Det BS' is either we used deterministic beam search to generate samples with very low perplexity. 'Mode' is about computing cosine similarities in K2T. 'max' means that for each token we select that token's maximum cosine similarity among each non appeared keyword. 'random' means that we randomly ranked keywords, and only calculate cosine similarity against the keyword which is currently considered. 'Gnte' means if we want hard constraint (guaranteeing the word appearance) or soft constraint.

5.3 Results

Table 2 represents our extensive experimentation for different settings. ID 12 shows the best performance, where ID 1 and 2 having the worst performance. See Discussion for comments on this table.

Table 3 represents our tests on the compiled 3400 high quality fake speeches. We used different random states to sample different real speeches from the validation set. Accuracy 0.5 means random guessing, thus our results prove that the successful BERT classifier actually confuses much against the final products of our pipeline.

Table 4 in Appendix shows some high quality examples with corresponding perplexities. Table 5 in Appendix shows weak generations of model ID 4 of 2 which are classified as fake by the classifier easily.

6 DISCUSSION

First, we will comment on Table 2. We can see that finetuning is actually critical for limiting the language model within the specific domain when we observe ID 1 and 2 performances. They produce weak performance for producing high quality speeches, but they are still able to fool the classifier for a few samples, probably because we give an initial prompt enough for them to produce occasional speech-like generations.

ID 5 has deterministic beam search as decoding, and it is among the worst performances. In this case, the generator produces texts with exceptional low perplexity, full with repetitions and simple phrases, which are easy to discriminate for classifier. Nucleus sampling is a remedy to this, which only samples from the subset of tokens holding the Top-p part of the probability mass.

Negative correlation exists between λ_0 and TPR. This means that shifting the output distribution too much towards the keywords can produce unrealistic text which are easy to detect. ID 6, 7, and 8 have higher λ_0 values, and they perform worse than the ones with smaller values.

As expected, initial prompt length holds high importance for the realistic generation. This task is easier than the one with no prompt because the language model will utilize from the provided natural text, and treat the rest of the generation as story completion. On the other hand, no prompt or prompt length 2 case needs to build the whole sentence from scratch, which sometimes can lead to unsound generations. ID 10, 11, and 12 shows this phenomenon.

In our experimentations, we do not see any clear effects of token count, mode, and gnte on the generation performance. This is somewhat expected because they have weak pressure on the generation performance.

Next, we will comment on Table 3, which is the most crucial table for telling the success of our pipeline. For any randomly sampled real texts, our synthetic compilation do not reveal itself as being fake from any perspective. This is apparent from the low values in any metrics.

Let us mention some drawbacks and limitations of our pipeline. K2T is a slow generation algorithm, thus we relied on sentences with max word count of 128. This does not pose any problem for our task as this length is still enough for expressing political opinions in text. We limited the number of keywords we use to generate sentences to 3 because increasing this for 128-word-sentences prevents the natural, and coherent generation.

7 FUTURE WORK

Our work opens the path for more robust and automatized fake generation pipelines. Potential directions of improvement can be enlisted as follows: Employing corpus specific keyword extractors, adapting K2T for fast generation, using language models with even higher number of parameters such as GPT-2 XLarge, using tailormade word embeddings specific to the domain we want to do generation (in our case, political word embeddings as in [10]), increasing corpus size, and evolving the current pipeline into a text-generator-model-agnostic quality control pipeline (plug-and-play generation step in Figure 1.)

8 CONCLUSION

In this work, we develop a method for automatizing the large scale generation of semantically conditioned domain specific realistic texts. Transformer based large pretrained models such as GPT-2 and BERT are the key players for generation and quality control steps, respectively. We showed the effectiveness of the method on U.S. Congressional Record Corpus, but the pipeline is flexible enough to be used for any specific domains.

REFERENCES

[1] David Ifeoluwa Adelani, Haotian Mai, Fuming Fang, Huy H Nguyen, Junichi Yamagishi, and Isao Echizen. 2020. Generating sentiment-preserving fake online reviews using neural language models and their human-and machine-based detection. In *International Conference on Advanced Information Networking and Applications*. Springer, 1341–1354.

[2] Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt. 2020. YAKE! Keyword extraction from single documents using multiple local features. *Information Sciences* 509 (2020), 257–289.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805 http://arxiv.org/abs/1810.04805

[4] Gloria Gennaro and Elliott Ash. 2022. Emotion and reason in political language. *The Economic Journal* 132, 643 (2022), 1037–1059.

[5] Rohit Kumar Kaliyar, Anurag Goswami, and Pratik Narang. 2021. FakeBERT: Fake news detection in social media with a BERT-based deep learning approach. *Multimedia tools and applications* 80, 8 (2021), 11765–11788.

[6] Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. 2019. CTRL: A Conditional Transformer Language Model for Controllable Generation. *CoRR* abs/1909.05858 (2019). arXiv:1909.05858 http://arxiv.org/abs/1909.05858

[7] Junyi Li, Tianyi Tang, Wayne Xin Zhao, and Ji-Rong Wen. 2021. Pretrained language models for text generation: A survey. *arXiv preprint arXiv:2105.10311* (2021).

[8] Xiang Lisa Li and Percy Liang. 2021. Prefix-Tuning: Optimizing Continuous Prompts for Generation. *CoRR* abs/2101.00190 (2021). arXiv:2101.00190 https://arxiv.org/abs/2101.00190

[9] Damian Pascual, Beni Egressy, Clara Meister, Ryan Cotterell, and Roger Wattenhofer. 2021. A Plug-and-Play Method for Controlled Text Generation. *CoRR* abs/2109.09707 (2021). arXiv:2109.09707 https://arxiv.org/abs/2109.09707

[10] Ludovic Rheault and Christopher Cochrane. 2020. Word Embeddings for the Analysis of Ideological Placement in Parliamentary Corpora. *Political Analysis* 28, 1 (2020), 112–133. https://doi.org/10.1017/pan.2019.26

[11] Congying Xia, Chenwei Zhang, Hoang Nguyen, Jiawei Zhang, and Philip S. Yu. 2020. CG-BERT: Conditional Text Generation with BERT for Generalized Few-shot Intent Detection. *CoRR* abs/2004.01881 (2020). arXiv:2004.01881 https://arxiv.org/abs/2004.01881

[12] Kevin Yang and Dan Klein. 2021. FUDGE: Controlled Text Generation With Future Discriminators. *CoRR* abs/2104.05218 (2021). arXiv:2104.05218 https://arxiv.org/abs/2104.05218

[13] Hanqing Zhang, Haolin Song, Shaoyu Li, Ming Zhou, and Dawei Song. 2022. A Survey of Controllable Text Generation using Transformer-based Pre-trained Language Models. *CoRR* abs/2201.05337 (2022). arXiv:2201.05337 https://arxiv.org/abs/2201.05337

A APPENDIX

A.1 Examples from Generated Texts

Speech	Prp
Mr. President, I am sending this to the desk on behalf of myself, Senator Abraham, and Senator Lieberman. It is a sense of Congress that states that do not have similar Medicare reductions should not incur substantial Medicare reductions, but States can at least reasonably assure that this will be the case. Medicare, the bill, would eliminate cuts to Medicare santorum proposal which is being considered by the Senate. To ensure that the Senator from Massachusetts will not see to it that I fail to mention his names, I send this message to the desk. The assassinated family of Aideen	65
Mr. President, I would like to announce for the public that a hearing has been scheduled before the Senate Committee on Energy and Natural Resources. The hearing will be held on Wednesday, September 18 at 10 a.m., in room SD-366 of the Dirksen Senate Office Building. The purpose of this hearing is to receive testimony on the President's Energy Plan for the 21st Century. Because of the limited time available for the hearings, witnesses may testify by invitation only. However, those wishing to submit written testimony for the hearing record should send two copies of their testimony to the Committee	15
Mr. President, I speak about the need for hate crimes legislation. On May 1, 2003, Senator Kennedy and I introduced the Local Law Enforcement Enhancement Act, a bill that would add new categories to current hate crimes law, sending a signal that violence of any kind is unacceptable in our society. On September 3, 2001, three men were found guilty of beating and shooting two women in their downtown Atlanta apartment for allegedly being gay and mentally retarded. Crimes motivated by race, gender, religion, sexual orientation, disability or national origin are routinely covered by State and local governments, as are	23
Mr. Chairman, I want to extend congratulations to the gentleman from California [Mr. Bono], my friend from Palm Springs, for the valiant effort he has put forth in developing the amendment which I am now offering today and very pleased to be joined by my friend and colleague, Chairman Thompson. Mr. Chairman, I yield the balance of my time to the gentleman from California [Mr. Bono].	45
Mr. President, I ask unanimous consent that the Subcommittee on Education, Arts and Humanities of the Committee on Labor and Human Resources be authorized to meet during the session of the Senate on March 18 at 10 a.m., to conduct a hearing entitled "Proposals for the National Institutes of Health." The	125
Mr. Speaker, 15 years ago last Monday I was a young television reporter in a small town called Saxonburg, PA, which now happens to be the home of the four Monday Night Football playoff games. The day before the game, the Houston Texans beat the Cincinnati Bengals, the only team that had not been to a playoff game since winning the Super Bowl in 1966. The win came against the Cincinnati Bengals on a Wednesday night because that was the only day that the FBI–because the fans gave the game away–also gave away the game. I don't think most people	28
Mr. Chairman, if the gentlewoman from Michigan is willing to accept the same deal that the gentleman from California just accepted, we will accept the same deal in return for which the Committee on Rules just voted in a bipartisan way for the National Park Service across-the-board cuts and the 15 percent more on the road money. Mr. Chairman, I yield to the gentleman from Utah [Mr. Hansen] for the purpose of accepting the gentleman's offer.	65
Mr. President, first, I remind my colleagues on both sides of the aisle that S. 770 is still at the desk and will be there for Members who want to offer amendments. I will make a unanimous consent request that will come later today so we can debate amendments on the bill tomorrow. I yield the floor and I suggest the absence of a quorum. The	99
Mr. President, I rise today in honor of Mr. Thomas "Buddy" Morgan, the general manager of the President's American Legion Post 61 in Rogers, AR, who passed away on December 4, 2006. Buddy was the second best man to perform the National Flag Gallantry Medal at the annual White House Ceremony, where the President selected those veterans who displayed the highest levels of heroism and gallantry while serving their country in uniform. As a young boy in Rogers, Buddy, then 17 years old, befriended an older soldier named Pete Delaney. At the young age of 13, Buddy helped Pete qualify	33
Mr. President, if there is nothing further to come before the Senate, I ask unanimous consent that the Senate stand in adjournment under the previous order. There being no objection, the Senate, at 8:50 p.m., adjourned until Wednesday, September 21, 1998, at 9:30 a.m.	146

Table 4: Random samples from high quality synthetic speech dataset. Perplexity (Prp) is rounded to nearest integer.

Speech	Prp
4230 south rio de la plata avenue in my district is not all that different from many others in the district i represent today because it is home to children first home of arizona. children first is a home for children that are in the foster care system who come here and have parents looking for them but have been turned away because they are too worried about how they will go back home and about how they will live their lives if they do go back home. children first home grandchildren are sometimes the only ones left. through nothing	37
4780 weeks since we first passed the women's health care package in a bipartisan way in the house of representatives. during that time more than 20 million women have gained the health care postpartum care coverage of women's health month. during that time nearly half a million fewer than 30 percent of women have been denied medical care because of their personal and medical history. and now it is time for the senate to get its work done to nuclearize the bill and adopt the h.r. 2642, the paul simon america's health plan–we don't need another gop-led congress to get	35
mr. chairman, this amendment would reduce by the same amount the increase in the president's request for administration and administration administration-related accounts authorized in this act by 30 million and would reduce the authorization of the congress for reconstruction by 30 million. the amendment is consistent with congressional priorities established in the republican contract on america, and i support its consideration. mr. chairman, i reserve the balance of my time	167
mr. president, i rise today to pay tribute to a good friend and colleague of mine from the great state of north carolina, who in his long and distinguished life has served our country and america admirably. senator thad cochran passed away on april 6, 2003, at the age of 79, and senator cochran's legislative record reflects his love of his state wellstone. he served his community in the same way that members of every south carolina congressional delegation did–serving his constituents and representing his state faithfully. senator cochran was born in cocobolo, nc,rewinesander cochanderene cochanderanda cochanderanda bou villalba.	60
mr. chairman, this amendment would prevent the use of funds for the national guard support operations or an equivalent program. many members of this body and many other members are very concerned with the military forces in this country, particularly the men and women who serve our nation overseas. we pay for them with our blood, sweat, and tears, not taxpayer dollars. my amendment would prohibit all funding for the national guard support operations, and it would prevent the department of defense from utilizing funds from hubbard hubbard hubbard hubbard hubbard hubbard hubbard hubbard hubbard hubbard hubbard hubbard hubbard hubbard	18
i have a big question for the senator on this issue that we will have time to talk about later this afternoon and throughout the afternoon as we have this important vote on the motion to proceed to the defense appropriations bill to fund the department of defense for fiscal year 1996 and for the years 1997 and 1998 as well as support for the critical job that we have done in america in all of these 10 years in the global war on terror. the senator has stated it well earlier today, and i am going to reiterate it	23
mr. speaker, what i ask is a short response. if my colleagues on the other side of the aisle are concerned about what we have done in the consideration of the balanced budget amendment and not about what we have done this morning, they should talk to me about this gentleman from florida or a question the president has asked, his remarks yesterday, or an hour ago	82

Table 5: Random fake samples produced by ID 4 in Table 2, which are classified as fake. Perplexity (Prp) is rounded to nearest integer. Lowercasing is due to us while saving the predictions. Note the repetitions, and grammatical errors.

A.2 Context Dependent Document Embeddings

Below are the previous ideas we have tried out at the first phase of the project. They mostly dealt with the problem of if it is possible to have context dependent embeddings documentwise. Please refer to the rough draft for more information.

A.2.1 Prefix-Tuning. Prefix-tuning [8] is the idea of learning task specific continuous prompts instead of natural language ones. By freezing the language model, and only learning small task specific prompt vector, it steers the generation better than sub-optimal description of the task used as prompt. It has shown effective for many NLP tasks such as table-to-text and summarization with low training cost than task specific fine-tuning.

We adopt this to our case as following: We see each speaker, party, state or combination of these such as party-state, party-parliament as different tasks whose vectors should be learnt during training. Furthermore, these learnt vectors might serve as document embeddings, enabling similarity comparison. After getting the embedding vector of the given speech's category, a linear layer is used for projecting it to a higher dimensionality. Then, the resulting vector is divided into a predefined number of tokens which will be prepended to given speech. The training task is again causal language modeling.

A.2.2 BERT-VAE. CG-BERT [11] combines BERT with VAE to generate new utterances for intent categories. For example, intent category is 'Alarm Query' and an utterance from this category is 'What are my alarms'. It aims to generate new utterances such as 'I need to see what alarms are there'.

The tokenized input ([CLS] + intent + [SEP] + utterance + [SEP]) is passed through first 6 layers of the pretrained BERT, and resulting CLS token is used to map corresponding sample to latent space. After sampling new CLS token, last 6 layers of BERT is passed to obtain generated sample. Here, since we sample CLS token midway, the generated sample will be slightly changed but still preserve the essence in utterance.

We extend this idea with some necessary changes. First, we are interested in clustering in latent space. As the above paper uses intent as condition, its model is a Conditional VAE, where each intent has its own latent space. Here, we will use plain VAE, where all speeches will be mapped into one latent space. The tokenized input to the model will be [CLS] speech [SEP]. Second, above paper does not make its implementation public. With HuggingFace library, it is not possible to do a forward pass only with last 6 layers, which prevents to sample CLS midway. A remedy to this is using two different BERTs: first as encoder, the second as decoder with LM head. The VAE is placed on top of output CLS token of first BERT.