TMLab: Generative Enhanced Model (GEM) for adversarial attacks

Piotr Niewiński, Maria Pszona, Maria Janicka
Samsung R&D Institute Poland
{p.niewinski, m.pszona, m.janicka}@samsung.com

Abstract

We present our Generative Enhanced Model (GEM) that we used to create samples awarded the first prize on the FEVER 2.0 *Breakers* Task. GEM is the extended language model developed upon GPT-2 architecture. The addition of novel target vocabulary input to the already existing context input enabled controlled text generation. The training procedure resulted in creating a model that inherited the knowledge of pretrained GPT-2, and therefore was ready to generate natural-like English sentences in the task domain with some additional control. As a result, GEM generated malicious claims that mixed facts from various articles, so it became difficult to classify their truthfulness.

1 Introduction

Fact-checking systems usually consist of separate modules devoted to information retrieval (IR) and recognizing textual entailment (RTE), also known as natural language inference (NLI). First, information retrieval module searches through the database in order to find sentences related to the given statement. Next, entailment module, with respect to the extracted sentences, classifies the given claim as TRUE, FALSE or NOT ENOUGH INFO. Currently, the best results are achieved by pretrained language models that are fine-tuned with task specific data (Yang et al., 2019; Liu et al., 2019).

Our task was to provide adversarial examples to break fact-checking systems. Since many fact-checking systems are based on neural language models, they might be less resistant to attacks with samples prepared within the same approach. In line with recent advances in natural language generation, we used the GPT-2 model (Radford et al., 2019), which we modified to prepare malicious

adversarial examples. GPT-2 generates subsequent sentences based on a given textual context and originally was trained on the WebText corpus. Our GEM architecture was expanded with *target* input for controlled generation, and carefully trained on the task data. During inference, the model was fed with the Wikipedia content. Simultaneously, *target* input was provided with named entities, terms and phrases extracted from Wikipedia articles.

2 FEVER *Breakers* Subtask

The second edition of Fact Extraction and Verification (FEVER 2.0) shared task was the three-phased contest utilizing the idea of adversarial training (Thorne and Vlachos, 2019). In the first phase, *Builders* had to create a fact-checking system. This system should extract evidence sentences for a given claim from Wikipedia articles that either SUPPORT or REFUTE this claim. It can also classify an example as NOT ENOUGH INFO. In the second phase, *Breakers* had to supply malicious examples to fool the existing systems. Finally, *Fixers* were obliged to improve those systems to withstand adversarial attacks. The model presented in this paper originated as a part of *Breakers* subtask. The aim of this task was to create adversarial examples that will break the majority of systems created in the *Builders* phase. Malicious claims could have been generated automatically or manually and were supposed to be balanced over three categories. The evidence sentences had to be provided in the SUPPORTS and REFUTES categories.

3 Generative Enhanced Model

3.1 Natural Language Generation with Neural Networks

Neural language models, such as GPT-2, rely on modeling conditional probability of an onco-

ming token for a given input sequence (context). Given the dictionary of tokens $\mathbb{D}$ and sequence $x_0 \ldots x_N$ ($x_i \in \mathbb{D}$) model computes conditional probability for every token x from $\mathbb{D}$:

$$p(x) = \prod_{i=1}^{n} p(x_n | x_1, \ldots, x_{n-1}).$$

During each stage of the process, the language model outputs probability distribution of tokens from dictionary $\mathbb{D}$. There are various approaches to select a single token from output distribution. Usually, the one with the highest probability is chosen or is sampled from the distribution. This distribution may be slightly modified by parameters like temperature and top-k. However, such context-based language generation gives us very little, if any, control over the model output.

Taking that into account, our main goal was to modify the architecture of Generative Pretraining Transformer (GPT-2), and enable additional control during the generation process. Therefore, GEM samples subsequent tokens by using information from two inputs: *context* (past) and *target*.

As target words various combinations of English nouns, verbs, and named entities can be provided and their number may vary.

GEM stops generating output when the total number of consecutive tokens reaches the value of parameter *maxTokens*. As a consequence, not only the first sentence, for which target words are given, is generated. That kind of generation procedure is expected to keep the original model's ability to build sentences even without target words. The examples of first sentences from the model output are presented in Table 1.

3.2 Architecture

GEM is build upon encoder-decoder Transformer-based language model architecture (Vaswani et al., 2017) enhanced with second Transformer encoder for *target* words.

Typical autoregressive neural language model, such as GPT-2, generates the next token using representations of *context* tokens (past) and *present* tokens (previously generated). Given *context* tokens $c_1 \ldots c_n$ and *present* tokens $p_1 \ldots p_n$,

context	The main railway stations of the province are Bydgoszcz and Toruń. Both stations are served by fast PKP Intercity trains which connect them with the capital Warsaw, as well as other major Polish cities.
target	**characterized Portugal farmland numerous lakes and forests**
output	Bydgoszcz is characterized as a medium-sized city in Portugal, with its farmland and numerous lakes and forests.
context	Near the beginning of his career, Einstein thought that Newtonian mechanics was no longer enough to reconcile the laws of classical mechanics with the laws of the electromagnetic field. This led him to develop his special theory of relativity during his time at the Swiss Patent Office in Bern (1902–1909).
target	**objected quantum mechanics contrast Bohr**
output	Einstein objected to the use of quantum mechanics in contrast with Bohr's theory of gravitation, which he thought was the most superior theory of relativity.
context	The City of New York, usually called either New York City (NYC) or simply New York (NY), is the most populous city in the United States. With an estimated 2018 population of 8,398,748 distributed over a land area of about 302.6 square miles (784 km2), New York is also the most densely populated major city in the United States.
target	**realized asset establishment independent border**
output	New York City is realized as an economic, cultural, and political asset upon the establishment of an independent border country.
context	Lasse Hoile (born 1973 in Aarhus, Denmark) is an artist, photographer and film-maker. He has collaborated with musician Steven Wilson and his projects Porcupine Tree and Blackfield. He has also designed live visuals for the US progressive metal band Dream Theater.
target	**true fact Swedish progressive metal band Stockholm**
output	Hoile's true interest is in fact the Swedish progressive metal band, Stockholm.

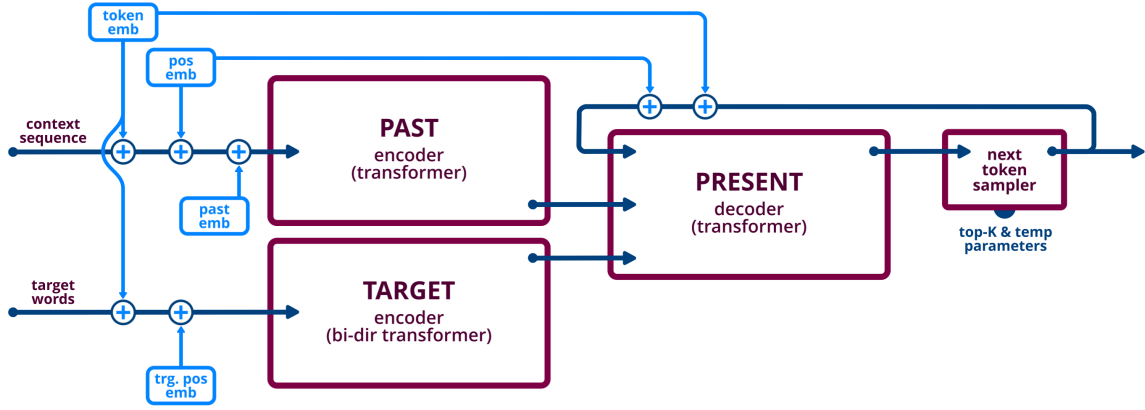Table 1: Examples of first sentences generated for given context and target words.

Figure 1: The architecture of GEM.

model encodes *context* representations $cr_1 \ldots cr_n$ and *present* representations $pr_1 \ldots pr_n$. With concatenated representations of *context* and *present* $cr_1 \ldots cr_n; pr1 \ldots pr_n$ model generates the next token p_{n+1}.

Both *context* and *present* representations are prepared with Transformers, using same shared parameters and embeddings. Such concept minimizes the number of parameters, and is optimal for classic generation task. Representations of *context* and *present* when concatenated are undifferentiated for decoder attention mechanism - the decoder has no information where *context* ends and *present* starts. This is not a problem for a standard task of neural language modeling.

The GEM's architecture is outlined in Figure 1. Just like GPT-2, the proposed model uses concatenated representations. However, in GEM *target* words representations $tr_1 \ldots tr_n$, prepared by target encoder, are added:

$$cr_1 \ldots cr_n; tr_1 \ldots tr_n; pr_1 \ldots pr_n.$$

In contrast to standard neural language models, GEM, in order to work properly, needs to differentiate between all three sources of representations. Both positional embeddings and Transformer weights of target encoder are not shared with past encoder and present decoder, and are initialized from scratch instead (with random normal initializer of 0.02 standard deviation). During the training, GEM learns the weights of target encoder to properly accomplish the task and distinguish *target* representations from the other two: *context* and *present*. In order to pass the information about the origin of *context* representations, we have ad-

ded *past* embedding (single trainable vector) to *context* tokens.

Past encoder and present decoder were initialized with GPT-2 checkpoint parameters. The idea was to use the knowledge of pretrained state-of-the-art English language model. Token embeddings from GPT-2 checkpoint were not updated while training. The final size of GEM is equal to 170-190% of the original GPT-2 model size (depending on GPT-2 version).

3.3 Training Procedure

The model was trained on the corpus provided by FEVER organizers. It contains a dump taken from the English-language version of Wikipedia from 2017. Each article was sentence-tokenized with spaCy tokenizer (Honnibal and Montani, 2017), and then each sentence was tokenized with BPE tokens from GPT-2 model.

Single training sample was prepared with the following procedure. First, random *target* sentence from the given Wikipedia articles was chosen. Next, the arbitrary number of words ranging from 20% to 60% was selected from *target* sentence. Selected words built *target* input. In addition, a small number of random words (up to 10%), which do not appear in *target* sentence, may be added to the set of *target* words. The intuition behind adding the noise to *target* words during the training phase was that it would prevent the model from directly 'copying' from *target* input. The model was supposed to decide whether to include the given words or not, because some of them may be irrelevant. Sentences forerunning *target* sentence established *context* input. As a result, *tar-*
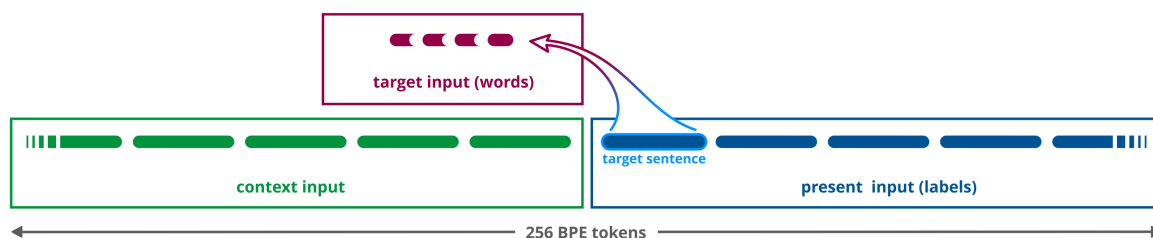
Figure 2: GEM training sample.

get sentence with the following sentences served as gold labels. In addition, single training sample was limited to 256 BPE tokens, which on average corresponds to 10 sentences. Single training sample is presented in Figure 2.

We have fine-tuned the original GPT-2 language model with the text generation task on the FEVER Wikipedia data (30M sentences). The model was fed with the Wikipedia content, and was asked to generate next sentences. GPT-2, without additional training, managed to achieve 37% accuracy on the stated task. It means that 37% of tokens generated by the model matched the gold labels from the original Wikipedia text. However, not modified GPT-2 fine-tuned with the given Wikipedia data was able to achieve 43% accuracy on a validation set.

Naturally, we expected higher accuracy with additional target words input. Though, we were afraid that adding new parameters and modifying the architecture might result in a significant loss of GPT-2 pretrained knowledge. During the training process, the initial accuracy of GEM was 3% and it raised very quickly. After the first epoch of training it achieved 47%. We trained the model with batches of 16 samples for 6 epochs, and the learning rate was set to 1e-5. The batch size of training data was limited by the memory of GPU, while other hyperparameters were chosen with the grid search evaluation. As a result, GEM finally achieved 53% accuracy while still not overfitting the data. High final accuracy of GEM states that the knowledge of GPT-2 was not forgotten, and, at the same time, the model learned to effectively use the provided target words.

We can estimate the theoretical maximum accuracy (higher bound) of GEM with stated task and training scheme. Each training sample, on average, corresponds to 10 sentences. The model generates tokens for 5 sentences. GEM additionally

fed with target words is able to achieve the maximum accuracy of 100% for the first sentence, and keep the maximum accuracy of fine-tuned GPT-2 (43%) for the remaining four sentences. With these assumptions, the average accuracy across the entire sample would reach 54.4%. Therefore, our final 53% accuracy is only a bit lower, and reversing these calculations we can get up to 93% accuracy for the first sentence when GEM is supported with target input.

4 Claims generation procedure

The procedure of generating claims was driven by the assumption that sophisticated claims contain knowledge from many sources, and cannot be checked with a single evidence sentence. To force automatic generation of such claims, we have built pipeline for input data preparation and claims selection described below. Wikipedia articles have a hypertext form with references to other articles. A single input sample (context and target words) was based on two Wikipedia articles: wiki-A and wiki-B. Wiki-A was randomly selected from the corpus. A set $\mathbb{B}$ was created from articles hyperlinked in the first five sentences of wiki-A. Then, it was filtered with the following principles. An article b was removed from $\mathbb{B}$ if:

- any words from title of b appeared in wiki-A title

- b hyperlink (string) in wiki-A was equal to b title

Finally, wiki-B article was randomly selected from $\mathbb{B}$.

The target words were randomly selected from the second sentence of wiki-B. Similar to the training procedure, their number varied from 20% to 60% of source-sentence words. Context sentences were composed of mixed wiki-A and wiki-B sentences, excluding sentences containing hyperlinks

23

to wiki-B and the second sentence of wiki-B. Finally, the title of wiki-A article was appended to the context. GEM started generation from this point.

Generated claims were further filtered, and the ones meeting any of the listed conditions were removed:

- claims not ending with a dot (probably due to incorrect tokenization)

- claims shorter than 30 characters and longer than 200

- claims containing `<endoftext>` token

- claims too similar to the first sentence of wiki-A (measured with Levenshtein (1966) distance)

- claims containing numbers and dates not appearing in wiki-A article

- claims containing any words out-of-vocabulary, where vocabulary was built from words of all Wikipedia articles

The examples of generated claims are shown in Table 2.

The dependency between the number of provided target words and the length of generated sentence is presented in Figure 3a. The statistics of target words number is shown in Figure 3b. The presented results are based on 1917 samples generated by GEM model and clearly indicates the correlation between the length of generated sentence and number of the target words: the fewer words the system gets, the shorter sentence will be generated.

The automatically generated claims required further manual labeling as SUPPORTS, REFUTES or NOT ENOUGH INFO. Moreover, in the case of the first two classes, the evidence senten-

wiki-A	Joseph Cao
wiki-B	Republican Party (United States)
context	Ánh Quang "Joseph" Cao ([ˈɡaʊ]; Cao Quang Ánh born March 13, 1967) is a Vietnamese American politician who was the U.S. Representative for from 2009 to 2011. In April 2011, Cao announced his candidacy for the office of Attorney General of Louisiana; however, in September 2011 he pulled out of the race, and the incumbent Buddy Caldwell ran unopposed for a second term. He is the first Vietnamese American to serve in Congress, and the first and thus far only Republican from his New Orleans-based district since 1891. In December 2015, he announced that he would run for the open U.S. Senate seat being vacated by retiring fellow Republican David Vitter in 2016. The Republican Party, commonly referred to as the GOP (abbreviation for Grand Old Party), is one of the two major contemporary political parties in the United States, the other being its historic rival, the Democratic Party. He is a member of the Republican Party.
target	**The party named dominant value during**
output	Joseph Cao was elected to Congress in 2009 and has named a number of prominent Republicans to be the dominant value players during his time in the House.
wiki-A	Remmina
wiki-B	Remote Desktop Protocol
context	Remmina is a remote desktop software client for POSIX-based computer operating systems. Remmina is in the package repositories for Debian versions 6 (Squeeze) and later and for Ubuntu versions since 10.04 (Lucid Lynx). As of 11.04 (Natty Narwhal), it replaced tsclient as Ubuntu's default remote desktop client. The FreeBSD ports/package collection also contains it as a separate port and additional protocol-specific plugin ports. Remote Desktop Protocol (RDP) is a proprietary protocol developed by Microsoft, which provides a user with a graphical interface to connect to another computer over a network connection. It supports the RDP, VNC, NX, XDMCP, SPICE and SSH protocols.
target	**RDP client for must run software**
output	Remmina is a standalone RDP client for Windows and must run as a user on a Linux system, or the client software will be unavailable.

Table 2: Examples of generated sentences for given context and target words.
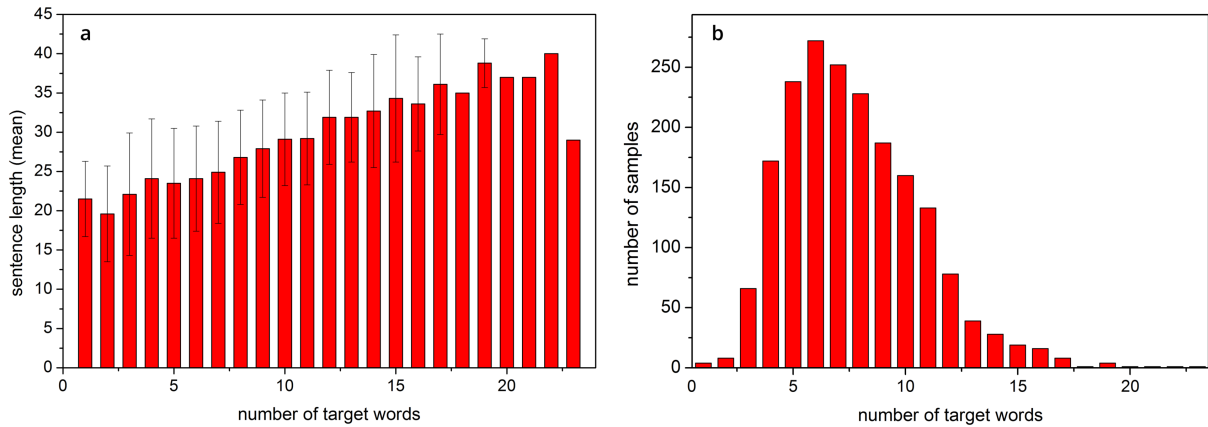
Figure 3: The dependency of the length of generated sentences on the number of target words (a) and target words statistics (b).

ces from Wikipedia were supposed to be delivered. Initially, each claim was annotated independently by two linguists. Both annotators agreed on 58.5% samples. The distribution of labels was highly unbalanced: 72.6% REFUTES, 13.7% NOT ENOUGH INFO, and 4.3% SUPPORTS. The remaining 9.4% of samples contained language errors. Finally, the supporting sentences from Wikipedia were manually extracted.

Due to a small number of claims labeled SUPPORTS in automatically generated data, there was a need to manually create some examples in this category. Malicious claims were based on several tricks, such as the usage of double negation, polysemy, comparison (of age, area, population), calculations, paraphrase (e.g. using phrases from Wikipedia articles unrelated to a claim or evidence), complex chains of reasoning, etc. The examples are provided in Table 3.

5 Results

Our adversarial attack was ranked the first place in the official FEVER 2.0 results. In total, we sub-

mitted 155 various claims (104 automatically generated and 51 written by human), which were divided into train and test sets. The quality of the test set was described by three measures: Correct Rate, Raw Potency and Potency, all defined in Thorne and Vlachos (2019). The Correct Rate, which is a percentage of positively verified samples, was 84.81%. This means that the organizers disqualified about 15% of our claims, mostly due to grammatical errors, such as word repetitions or wrong verb forms. The Raw Potency of the prepared adversarial examples, defined as the percentage of incorrect predictions, averaged over all systems was 78.80%. Finally, the main evaluation measure - Potency (the Raw Potency scaled by the Correct Rate) achieved by our samples was 66.83%.

6 Conclusions

The claims provided by GEM model appeared to be the most challenging for fact-checking systems competing in a FEVER 2.0 shared task. Our strategy was to mix Wikipedia articles, which were

double negation	It is not true that one can falsely say that double negation theorem states that "If a statement is false, than it is not the case that the statement is not false."
comparison	Łączka does not lay as close to Siedlce as Żuków.
paraphrase	Finding a theory of everything, which is considered a final theory, still remains a challenge.
polysemy	There is a fashion house with a word meaning 'sweet' in its name.
negation	K2 is not the highest mountain in the world.

Table 3: Examples of manually prepared samples.

connected to each other with a hyperlink and filtered with the established strategies. This approach led to generating cohesive, well-structured samples, which were challenging for automated verification. As GEM was developed upon GPT-2 architecture, and inherited its knowledge, the model might be biased towards factual inaccuracies. The established pipeline could just strengthen this tendency, which finally reflected in the class imbalance of automatically generated content. Automatic generation of complex claims supported by Wikipedia would require fine-tuned procedures. This issue seems to be an interesting challenge that could be addressed in further research.

The preparation of adversarial examples is a very prominent concept of modern machine learning research area. It gives the possibility of fast, automated, and massive generation of additional samples. Importantly, injecting the malicious examples into training data may result in more robust and accurate models. GEM designed for controlled text generation can also be applied in various text-driven systems, e.g. conversational agents, text summarizers or style transfer models.

References

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

VI Levenshtein. 1966. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10:707.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

James Thorne and Andreas Vlachos. 2019. Adversarial attacks against fact extraction and verification. *CoRR*, abs/1903.05543.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *CoRR*, abs/1906.08237.