Virus Escape Mutations
Learning the language of viral evolution and escape[Hie et al., 2021]

Shuyang Zhang
CS 886
University of Waterloo

February 11, 2022
1. Introduction

2. Constrained Semantic Change Search (CSCS)

3. Experiments and Results
   3.1 Natural Language - News Headlines
   3.2 Influenza, HIV and COVID-19

4. Conclusion
• Mutations happen in molecular sequences all the time. Some have little impact, some can cause serious issues.
Virus Escape Mutations

- Mutations happen in molecular sequences all the time. Some have little impact, some can cause serious issues.
- Viruses mutate frequently. We want to look at how a virus can mutate to become more dangerous.
Virus Escape Mutations

- Mutations happen in molecular sequences all the time. Some have little impact, some can cause serious issues.
- Viruses mutate frequently. We want to look at how a virus can mutate to become more dangerous.
- How do we define “dangerous”? It has to be both viral (infectious) and escapes our human immune defense system.
Virus Escape Mutations Continued

Figure: How Neutralizing Human Antibodies block viruses from infecting cells
The Problem

- Our goal is to mutate the protein sequence of a virus such that it has antigenic change without losing viral infectivity.
The Problem

- Our goal is to mutate the protein sequence of a virus such that it has antigenic change without losing viral infectivity.
- We can compare this to a natural language task, where we mutate a sentence such that it changes its meaning (semantic change), while keeping it still grammatical.
The Problem

- Our goal is to mutate the protein sequence of a virus such that it has antigenic change without losing viral infectivity.
- We can compare this to a natural language task, where we mutate a sentence such that it changes its meaning (semantic change), while keeping it still grammatical.
- In the presentation I will be introducing a solution for this problem: **Constrained Semantic Change Search** (CSCS).
The Problem

- Our goal is to mutate the protein sequence of a virus such that it has antigenic change without losing viral infectivity.
- We can compare this to a natural language task, where we mutate a sentence such that it changes its meaning (semantic change), while keeping it still grammatical.
- In the presentation I will be introducing a solution for this problem: **Constrained Semantic Change Search (CSCS).**

**Figure:** CSCS objective: increasing semantic change while maintaining high grammaticality
Here is an example of grammatical semantic change in English.

Original: nauru bans transhipments to tackle overfishing
Semantically closest: nauru bans transhipments to combat overfishing
CSCS change: nauru bans continue to tackle overfishing
1. Introduction

2. Constrained Semantic Change Search (CSCS)

3. Experiments and Results
   3.1 Natural Language - News Headlines
   3.2 Influenza, HIV and COVID-19

4. Conclusion
We are given a sequence of tokens $x = (x_1, x_2, \ldots, x_N)$, $x_i \in \mathcal{X}$. $\mathcal{X}$ is a finite alphabet.
Formulating the Problem - CSCS

We are given a sequence of tokens $\mathbf{x} = (x_1, x_2, \ldots, x_N)$, $x_i \in \mathcal{X}$. $\mathcal{X}$ is a finite alphabet. Let $\tilde{x}_i$ be a mutation at position $i$ and $\mathbf{x}[\tilde{x}_i] = (\ldots, x_{i-1}, \tilde{x}_i, x_{i+1}, \ldots)$. 

Let $\tilde{x}_i$ be a mutation at position $i$ and $\mathbf{x}[\tilde{x}_i] = (\ldots, x_{i-1}, \tilde{x}_i, x_{i+1}, \ldots)$. 

On the other hand, grammaticality of a mutation is given by $p(\tilde{x}_i | \mathbf{x})$, which takes value close to 0 if $\mathbf{x}[\tilde{x}_i]$ is not grammatical and 1 otherwise. 

For each mutation, we compute the following term $a(\tilde{x}_i; \mathbf{x}) = \Delta z[\tilde{x}_i] + \beta p(\tilde{x}_i | \mathbf{x})$, where the hyperparameter $\beta \in [0, \infty)$ and we use this term to rank mutations.
Formulating the Problem - CSCS

We are given a sequence of tokens $x = (x_1, x_2, \ldots, x_N)$, $x_i \in \mathcal{X}$. $\mathcal{X}$ is a finite alphabet. Let $\tilde{x}_i$ be a mutation at position $i$ and $x[\tilde{x}_i] = (\ldots, x_{i-1}, \tilde{x}_i, x_{i+1}, \ldots)$. Let $f_s : \mathcal{X}^N \rightarrow \mathbb{R}^k$ be a semantic embedding. Then we define semantic change as

$$\Delta z[\tilde{x}_i] = \|f_s(x) - f_s(x[\tilde{x}_i])\|$$

(1)
Formulating the Problem - CSCS

We are given a sequence of tokens $x = (x_1, x_2, \ldots, x_N)$, $x_i \in \mathcal{X}$. $\mathcal{X}$ is a finite alphabet.

Let $\tilde{x}_i$ be a mutation at position $i$ and $x[\tilde{x}_i] = (\ldots, x_{i-1}, \tilde{x}_i, x_{i+1}, \ldots)$.

Let $f_s : \mathcal{X}^N \rightarrow \mathbb{R}^k$ be a semantic embedding. Then we define semantic change as

$$\Delta z[\tilde{x}_i] = \|f_s(x) - f_s(x[\tilde{x}_i])\| \quad (1)$$

On the other hand, grammaticality of a mutation is given by

$$p(\tilde{x}_i | x), \quad (2)$$

which takes value close to 0 if $x[\tilde{x}_i]$ is not grammatical and 1 otherwise.
We are given a sequence of tokens $x = (x_1, x_2, \ldots, x_N)$, $x_i \in \mathcal{X}$. $\mathcal{X}$ is a finite alphabet.

Let $\tilde{x}_i$ be a mutation at position $i$ and $x[\tilde{x}_i] = (\ldots, x_{i-1}, \tilde{x}_i, x_{i+1}, \ldots)$.

Let $f_s : \mathcal{X}^N \rightarrow \mathbb{R}^k$ be a semantic embedding. Then we define semantic change as

$$\Delta z[\tilde{x}_i] = \|f_s(x) - f_s(x[\tilde{x}_i])\|$$

On the other hand, grammaticality of a mutation is given by

$$p(\tilde{x}_i|x)$$

which takes value close to 0 if $x[\tilde{x}_i]$ is not grammatical and 1 otherwise.

For each mutation, we compute the following term

$$a(\tilde{x}_i; x) = \Delta z[\tilde{x}_i] + \beta p(\tilde{x}_i|x),$$

where the hyperparameter $\beta \in [0, \infty)$ and we use this term to rank mutations.
Building the Language Model

Given our objectives defined above, how do we actually get the semantic embedding $f_s$ and the grammaticality function $p$?
Building the Language Model

Given our objectives defined above, how do we actually get the semantic embedding \( f_s \) and the grammaticality function \( p \)?

We look at the full sequence context of a token \( x_{[N]\{x_i} \), and build an architecture with two-stacked BiLSTM layers instantiating the semantic embedding function \( \hat{f}_s \), such that

\[
\hat{z}_i = \hat{f}_s(x_{[N]\{i}) = [\text{LSTM}_f(g_f(x_1, \ldots, x_{i-1}))^T \text{ LSTM}_r(g_r(x_{i+1}, \ldots, x_N))^T]^T,
\]

and a latent variable probability \( \hat{p}(x_i|x_{[N]\{i}; \hat{z}_i) = \hat{p}(x_i|\hat{z}_i) = \text{softmax}(W\hat{z}_i + b) \).
Building the Language Model

Given our objectives defined above, how do we actually get the semantic embedding $f_s$ and the grammaticality function $p$?

We look at the full sequence context of a token $x_{[N]\{x_i\}}$, and build an architecture with two-stacked BiLSTM layers instantiating the semantic embedding function $\hat{f}_s$, such that

$$\hat{z}_i = \hat{f}_s(x_{[N]\{i\}}) = [\text{LSTM}_f(g_f(x_1, \ldots, x_{i-1}))^T \text{LSTM}_r(g_r(x_{i+1}, \ldots, x_N))^T]^T,$$

and a latent variable probability $\hat{p}(x_i|x_{[N]\{i\}}; \hat{z}_i) = \hat{p}(x_i|\hat{z}_i) = \text{softmax}(W\hat{z}_i + b)$.

Here since $\hat{z}_i$ encode the context of $x_i$, then given $\hat{z}_i$, $x_i$ is conditionally independent of its context.
Building the Language Model

Given our objectives defined above, how do we actually get the semantic embedding $f_s$ and the grammaticality function $p$?

We look at the full sequence context of a token $x_{[N]\{x_i\}}$, and build an architecture with two-stacked BiLSTM layers instantiating the semantic embedding function $\hat{f}_s$, such that

$$\hat{z}_i = \hat{f}_s(x_{[N]\{i\}}) = [\text{LSTM}_f(g_f(x_1, \ldots, x_{i-1}))^T \text{LSTM}_r(g_r(x_{i+1}, \ldots, x_N))^T]^T,$$

and a latent variable probability $\hat{p}(x_i|x_{[N]\{i\}}; \hat{z}_i) = \hat{p}(x_i|\hat{z}_i) = \text{softmax}(W\hat{z}_i + b)$.

Here since $\hat{z}_i$ encode the context of $x_i$, then given $\hat{z}_i$, $x_i$ is conditionally independent of its context.

We define $\hat{z}$ as the concatenation of the semantic embeddings of each word $\hat{z}_i$. 
Building the Language Model

Given our objectives defined above, how do we actually get the semantic embedding $f_s$ and the grammaticality function $p$?

We look at the full sequence context of a token $x_{[N]\{x_i\}}$, and build an architecture with two-stacked BiLSTM layers instantiating the semantic embedding function $\hat{f}_s$, such that

$$\hat{z}_i = \hat{f}_s(x_{[N]\{i\}}) = [\text{LSTM}_f(g_f(x_1, \ldots, x_{i-1}))^T \text{LSTM}_r(g_r(x_{i+1}, \ldots, x_N))^T]^T,$$

and a latent variable probability $\hat{p}(x_i|x_{[N]\{i\}}; \hat{z}_i) = \hat{p}(x_i|\hat{z}_i) = \text{softmax}(W\hat{z}_i + b)$.

Here since $\hat{z}_i$ encode the context of $x_i$, then given $\hat{z}_i$, $x_i$ is conditionally independent of its context.

We define $\hat{z}$ as the concatenation of the semantic embeddings of each word $\hat{z}_i$, so

$$\Delta z[\hat{x}_i] = \|\hat{z} - \hat{z}[\hat{x}_i]\|_1, \ p(\hat{x}_i|x) = \hat{p}(\hat{x}_i|\hat{z}_i)$$
Language Model Illustration

\[ \hat{p}(x_i | \hat{z}_i) \]
\[ \hat{z}_i = \hat{f}(x_{[N]\{i\}}) \]

\[ A \quad \cdots \quad I \quad \cdots \quad Y \]
\[ 0.1 \quad \cdots \quad 0.2 \quad \cdots \quad 0.05 \]

\[ x_{[N]\{i\}} \]
\[ V \quad L \quad S \quad _{\_} \quad K \quad A \quad A \]

**Figure:** Graph Description of the CSCS model with two stacked BiLSTM layers
Instead of adding the semantic change and grammaticality scores together directly, we apply rank-based acquisition.

\[ a'(\tilde{x}_i; x) = \text{rank}(\Delta z[\tilde{x}_i]) + \beta \text{rank}(p(\tilde{x}_i| x)) \]

For each mutation \(\tilde{x}_i\), we can then use \(a'\) to determine the priority of the mutation.
Instead of adding the semantic change and grammaticality scores together directly, we apply rank-based acquisition.

\[ a'(\tilde{x}_i; x) = \text{rank}(\Delta z[\tilde{x}_i]) + \beta \text{rank}(p(\tilde{x}_i|x)) \]

For each mutation \( \tilde{x}_i \), we can then use \( a' \) to determine the priority of the mutation.

The reason we use rank-based acquisition is because it makes scaling with \( \beta \) much easier and in practice, setting \( \beta = 1 \) achieves good performance.
Assumptions

There are some assumptions in our model.

- In the natural language case, antonyms (words with opposite meanings) may also be close in the semantic space. Similarly in the case for proteins, the semantic embedding may not perfectly model antigenic change.
- The grammaticality model may include more than grammar, such as pragmatics.
1. Introduction

2. Constrained Semantic Change Search (CSCS)

3. Experiments and Results
   3.1 Natural Language - News Headlines
   3.2 Influenza, HIV and COVID-19

4. Conclusion
### Table 1: Headline Semantic Change Results.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Median % POS Change NLTK</th>
<th>Median % POS Change FLAIR</th>
<th>Median WordNet Similarity Pathwise</th>
<th>Median WordNet Similarity Wu-Palmer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semantically closest (smallest $\Delta z[\tilde{x}_i]$)</td>
<td>0.00%</td>
<td>0.00%</td>
<td>0.143</td>
<td>0.546</td>
</tr>
<tr>
<td>CS-CS-proposed (highest $a^T(\tilde{x}_i; x)$)</td>
<td>16.7%</td>
<td>14.3%</td>
<td>0.0833</td>
<td>0.235</td>
</tr>
<tr>
<td>two-sided $t$-test $P$</td>
<td>$&lt;10^{-308}$</td>
<td>$&lt;10^{-308}$</td>
<td>$&lt;10^{-308}$</td>
<td>$&lt;10^{-308}$</td>
</tr>
</tbody>
</table>

### Table 2: Grammatical Acceptability Results

<table>
<thead>
<tr>
<th>Setting</th>
<th>Number Acceptable (Out of 300)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Human 1</td>
</tr>
<tr>
<td>CS-CS-proposed ($\beta = 0.25$)</td>
<td>130</td>
</tr>
<tr>
<td>CS-CS-proposed ($\beta = 1$)</td>
<td>200</td>
</tr>
<tr>
<td>Original headline</td>
<td>223</td>
</tr>
</tbody>
</table>
Influenza

Training data consists of 44,999 unique Influenza A hemogglutinin (HA) amino acid sequences observed in animal hosts. Since the data were all acquired from infected animal hosts, we know that they are already viral and infectious (grammatical).
Influenza

Training data consists of 44,999 unique Influenza A hemogglutinin (HA) amino acid sequences observed in animal hosts. Since the data were all acquired from infected animal hosts, we know that they are already viral and infectious (grammatical).

Visualizing the semantic embedding in two dimensions using UMAP, we see that different subtypes of the HA protein are clearly separated.
Influenza

Training data consists of 44,999 unique Influenza A hemogglutinin (HA) amino acid sequences observed in animal hosts. Since the data were all acquired from infected animal hosts, we know that they are already viral and infectious (grammatical).

Visualizing the semantic embedding in two dimensions using UMAP, we see that different subtypes of the HA protein are clearly separated.

Figure: Visualization of Semantic Mapping for Influenza Subtypes
Influenza Escape

To predict virus escape, they use a dataset by [Lee et al., 2019] where they assessed all possible single-residue mutations to the H3N2 strain and which mutant preserve viral infectivity and induce escape.
Influenza Escape

To predict virus escape, they use a dataset by [Lee et al., 2019] where they assessed all possible single-residue mutations to the H3N2 strain and which mutant preserve viral infectivity and induce escape.

Ensuring that none of the mutations in Lee’s database were present in the training data (Zero-Shot Prediction), Hie et al. used the CSCS model to predict mutations, and 4 out of 5 predicted mutants were confirmed escapes by Lee.
Influenza Escape

To predict virus escape, they use a dataset by [Lee et al., 2019] where they assessed all possible single-residue mutations to the H3N2 strain and which mutant preserve viral infectivity and induce escape.

Ensuring that none of the mutations in Lee’s database were present in the training data (Zero-Shot Prediction), Hie et al. used the CSCS model to predict mutations, and 4 out of 5 predicted mutants were confirmed escapes by Lee.

Figure: CSCS-predictions correspond to escape mutations with high semantic change and high grammaticality
Enrichment of Acquired Escapes

Another measurement of performance is the enrichment of acquired escapes, which is the area under the curve (AUC) of the plot with number of escapes versus total predicted mutations acquired by prioritizing $a'(\tilde{x}_i; x)$. 
Enrichment of Acquired Escapes

Another measurement of performance is the enrichment of acquired escapes, which is the area under the curve (AUC) of the plot with number of escapes versus total predicted mutations acquired by prioritizing $a'(\tilde{x}_i; x)$.

Figure: Escape mutants are enriched in CSCS-proposed mutations
Analyzing CSCS’s performance for HIV is quite similar to that for Influenza. Working with HIV-I Envelope (Env) proteins, Hie et al. first trained the model to learn semantic embedding and grammaticality.
Analyzing CSCS’s performance for HIV is quite similar to that for Influenza. Working with HIV-I Envelope (Env) proteins, Hie et al. first trained the model to learn semantic embedding and grammaticality.

They then performed zero-shot prediction on a database with infectivity and escape potential for all single-residue mutations for a strain of HIV. Enrichment of acquire escapes were less obvious than that for HA, but still surpasses that of other approaches.
SARS-CoV-2 Regional Enrichment

Figure: Potential for Escape for COVID Spike protein has different levels in different regions.
Contents

1. Introduction

2. Constrained Semantic Change Search (CSCS)

3. Experiments and Results
   3.1 Natural Language - News Headlines
   3.2 Influenza, HIV and COVID-19

4. Conclusion
Related Work

• Viral mutation studies for specific viruses:
  • seasonal Influenza viruses ([Bedford et al., 2015])
  • SARS-CoV-2 ([Starr et al., 2021, Maher, 2021])
• Seq2seq GAN network for mutation generation ([Berman et al., 2020])
Conclusion

Hie et al.’s paper was the first to use both semantic change and grammaticality as measures for virus escape mutations. It presented a novel approach in predicting how a virus can mutate to escape the immune system and thus become more dangerous.

There still remains an ethical question: if used wrongly, will being able to predict virus escape mutations be a menace to our society?
Conclusion

Hie et al.’s paper was the first to use both semantic change and grammaticality as measures for virus escape mutations. It presented a novel approach in predicting how a virus can mutate to escape the immune system and thus become more dangerous.

They experimented on natural language as well as virus antigen proteins and showed significant enrichment in acquired escapes.
Hie et al.’s paper was the first to use both semantic change and grammaticality as measures for virus escape mutations. It presented a novel approach in predicting how a virus can mutate to escape the immune system and thus become more dangerous.

They experimented on natural language as well as virus antigen proteins and showed significant enrichment in acquired escapes.

Their research can help us better learn the mutations and variants of COVID-19 and other viruses to come, and be more prepared for future variants.
Conclusion

Hie et al.’s paper was the first to use both semantic change and grammaticality as measures for virus escape mutations. It presented a novel approach in predicting how a virus can mutate to escape the immune system and thus become more dangerous.

They experimented on natural language as well as virus antigen proteins and showed significant enrichment in acquired escapes.

Their research can help us better learn the mutations and variants of COVID-19 and other viruses to come, and be more prepared for future variants.

There still remains an ethical question: if used wrongly, will being able to predict virus escape mutations be a menace to our society?
B. Hie et al. (2021)
Learning the language of viral evolution and escape.
Science 371.6526 (2021): 284-288

T. Bedford et al. (2015)
Global circulation patterns of seasonal influenza viruses vary with antigenic drift
Nature 523(7559):217–220

M. Maher et al. (2021)
Predicting the mutational drivers of future SARS-CoV-2 variants of concern
medRxiv 2021.06.21.21259286; doi: https://doi.org/10.1101/2021.06.21.21259286
T. Starr et al. (2021)
Prospective mapping of viral mutations that escape antibodies used to treat COVID-19
*Science* 371(6531):850-854

D. Berman et al. (2020)
MutaGAN: A Seq2seq GAN Framework to Predict Mutations of Evolving Protein Populations

J. Lee et al. (2019)
Mapping person-to-person variation in viral mutations that escape polyclonal serum targeting influenza hemagglutinin
*Elife* 8:e49324.
Q & A