1. Wykorzystane gramatyki

W eksperymentach zdecydowano się na wykorzystanie gramatyk opisujących dwa języki które zostały przedstawione w artykule: „*GA-based Learning of Context-Free Grammars using Tabular Representations.*” Yasubumiego Sakakibariego oraz Mitsuhirego Kondo.[], oraz jedną gramatykę opisującą białka.

* 1. Pierwsza gramatyka testowa

Pierwsza gramatyka opisuje język: nad alfabetem: {a, b, c}.

Oznacza to, że wszystkie słowa należące do tego języka składają się z symboli: „a”, „b” oraz „c”, ułożonych w następującej sekwencji:

* na początku każdego słowa znajduje się *n* powtórzeń symbolu „a”,
* po symbolach „a” znajduje się taka sama liczba symboli „b”,
* każde słowo zakończone jest dowolną, niezerową liczbą powtórzeń symbolu „c”.

Utworzona na potrzeby eksperymentu gramatyka opisująca ten język, w postaci normalnej Chomsky’ego, składała się z:

* 3 symboli terminalnych: {a, b, c},
* 6 symboli nieterminalnych: {S, T, V, A, B, C}, spośród których symbol „S” jest symbolem startowym,
* 8 reguł: {S → TC, S → SC, T → AV, T → AB, V → TB, A → a, B → b, C → c}.

Jak widać, w przyjętej gramatyce, żaden z symboli nieterminalnych, zdolnych do generowania symbolu terminalnego nie jest zdolny do generowania innego symbolu nieterminalnego, natomiast żaden z symboli nieterminalnych zdolnych do generowania innego symbolu nieterminalnego nie jest zdolny do generowania jakiegokolwiek symbolu terminalnego.

Ze względu na to, symbole nieterminalne zostały podzielone na:

* symbole leksykalne (symbole nieterminalne generujące symbole terminalne): {A, B, C},
* symbole strukturalne (symbole nieterminalne generujące symbole nieterminalne): {S, T, V}.

Analogiczny podział zastosowano w przypadku reguł, które zostały podzielone na reguły leksykalne (generujące symbole terminalne) oraz reguły strukturalne (generujące symbole nieterminalne).

Na potrzeby maszynowego uczenia, dla powyższej gramatyki, wygenerowano gramatykę pokrywającą, czyli gramatykę zawierającą reguły umożliwiające generowanie dowolnego zestawu symboli nieterminalnych z każdego symbolu strukturalnego oraz każdego symbolu terminalnego z dowolnego symbolu leksykalnego.

Gramatyka pokrywająca została dodatkowo rozszerzona o jeden dodatkowy symbol strukturalny: „U” w związku z czym zawierała 9 reguł leksykalnych oraz 196 reguł strukturalnych.

Gramatyka ta będzie w dalszej części pracy nazywana pierwszą gramatyką testową, natomiast opisana wcześniej gramatyka przez nią rozszerzana: pierwszą gramatyką docelową.

* 1. Druga gramatyka testowa

Druga gramatyka opisuje język: nad alfabetem: {a, b, c}.

Oznacza to, że wszystkie słowa należące do tego języka składają się z symboli: „a” i „c” lub „b” i „c”, ułożonych w następującej sekwencji:

* na początku każdego słowa znajduje się jedno powtórzenie symbolu „a” lub „b”
* po pierwszym symbolu występuje dowolna, niezerowa liczba powtórzeń symbolu „c”.

Utworzona na potrzeby eksperymentu gramatyka opisująca ten język, w postaci normalnej Chomsky’ego, składała się z:

* 3 symboli terminalnych: {a, b, c},
* 3 symboli nieterminalnych: {S, T, C}, spośród których symbol „S” jest symbolem startowym,
* 5 reguł: {S → TC, S → SC, T → a, T → b, C → c}.

Podobnie jak w przypadku pierwszej gramatyki testowej i docelowej, symbole nieterminalne oraz reguły zostały podzielone na strukturalne oraz leksykalne.

Dla powyższej gramatyki, nazywanej od tej pory drugą gramatyką docelową, wygenerowano 4 rozszerzające je gramatyki pokrywające:

* do pierwszej z nich nie dodano żadnych dodatkowych symboli nieterminalnych,
* do drugiej dodano dodatkowy nieterminalny symbol strukturalny: „U”,
* do trzeciej dodano dodatkowy nieterminalny symbol leksykalny: „A”,
* do czwartej dodano dodatkowy nieterminalny symbol strukturalny: „U”, oraz dodatkowy nieterminalny symbol leksykalny: „A”,

Gramatyki te będą od teraz nazywane kolejno: „drugą gramatyką testową 1”, „drugą gramatyką testową 2”, „drugą gramatyką testową 3” oraz „drugą gramatyką testową 4”.

* 1. **Gramatyka opisująca białka**

No tutaj właśnie problem bo gdzieś mi się zapodział ten link do Pana artykułu, mógłbym prosić o ponowne podesłanie go?

1. **Przebieg eksperymentu**
   1. **Wpływ wartości hiperparametrów algorytmu maszynowego uczenia na jego przebieg i wyniki**

W pierwszej części eksperymentu zbadano to, w jaki sposób zmiana wartości hiperparametrów algorytmu uczącego wpływa na przebieg nauki i jej końcowe wyniki.

Sprawdzanymi hiperparmatrami funkcji była: liczba osobników w populacji (n), prawdopodobieństwo mutacji (pm), maksymalna skala mutacji (sm), oraz prawdopodobieństwo krzyżowania (pk).

Podczas eksperymentu wykonano szereg symulacji z różnymi zestawami hiperparametrów, wykorzystane zestawy przedstawiono w poniższej tabeli:

Tabela 2.1- wykorzystane zestawy hiperparametrów

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| l.p. | n | pm | ms | pk |
| 1 | 40 | 0,001 | 0,5 | 0,5 |
| 2 | 80 | 0,0001 | 0,5 | 0,5 |
| 3 | 80 | 0,001 | 0,5 | 0,5 |
| 4 | 80 | 0,01 | 0,5 | 0,5 |
| 5 | 80 | 0,1 | 0,5 | 0,5 |
| 6 | 80 | 0,001 | 0,1 | 0,5 |
| 7 | 80 | 0,001 | 0,9 | 0,5 |
| 9 | 80 | 0,001 | 0,5 | 0,1 |
| 10 | 80 | 0,001 | 0,5 | 0,9 |
| 11 | 160 | 0,0001 | 0,5 | 0,5 |
| 12 | 160 | 0,001 | 0,5 | 0,5 |
| 13 | 160 | 0,01 | 0,5 | 0,5 |
| 14 | 160 | 0,1 | 0,5 | 0,5 |
| 15 | 160 | 0,001 | 0,1 | 0,5 |
| 16 | 160 | 0,001 | 0,9 | 0,5 |
| 17 | 160 | 0,001 | 0,5 | 0,1 |
| 18 | 160 | 0,001 | 0,5 | 0,9 |
| 19 | 320 | 0,001 | 0,5 | 0,5 |