פרויקט בביולוגיה

Naor cohen & Ofir tanami

בפרויקט זה עבדנו עם רצפי חלבון RBNS כאשר לכל חלבון יש מספר קבצים המייצגים ריכוזים שונים שלו ובכל קובץ יש רצפי דנ"א שנקשרו לאותו חלבון בריכוז המסוים. הקבצים מיוצגים בפורמט הבא:

$$< input >< RBNS1 > \cdots < RBNS5 >$$

כאשר הקובץ input מייצג רצפי דנ"א בלי ריכוז חלבון וקובץ RBNS5 המייצג את הריכוז הכי גבוה של חלבון מסוים שהוכנס ומה שביניהם זה ריכוזים שונים בסדר עולה.

מטרתנו הייתה לאמן מודל בעזרת קבצים אלו לבחירתנו ולבחון את המודל המאומן על סט הטסט המכיל את רצפי ה RNCMPT המכילים רצפי דנ"א ולהחזיר את ההסתברות קשירה של אותם רצפים פר חלבון.

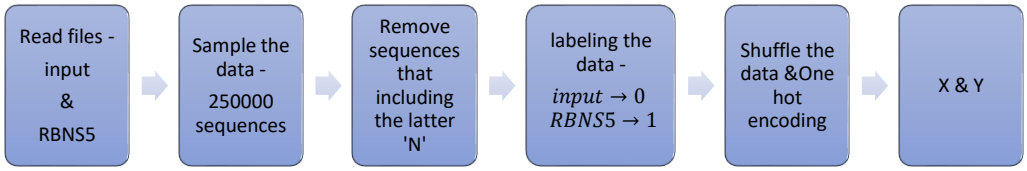
כעת נסביר על שלבי העבודה במודל שלנו :

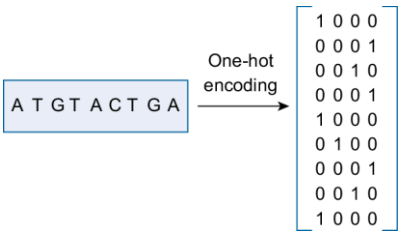
שלב 1- עיבוד מקדים:

אנו בחרנו לעבוד עם קובץ ה input המייצג כאמור רצפי דנ"א לפני שהוכנס חלבון בריכוז מסוים ובקובץ RBNS5 המייצג רצפי דנ"א אשרר נקשרו לחלבון בריכוז הכי גבוה. לאחר מכן לקחנו מכל קובץ 250000 דגימות ומאותן דגימות סיננו רצפים אשר מכילים את האות N והוספנו תיוגים באופן הבא :

$$input \rightarrow 0\ , RBNS5 \rightarrow 1$$

כלומר תייגנו את הקובץ בלי ריכוז חלבון כ 0 ואת הריכוז הכי גובה כ 1 או במילים אחרות, התייחסנו לרצפי הדנ"א בקובץ ה input כרצפים שלא נקשרו לחלבון ולקבצי הדנ"א בקובץ עם הריכוז הכי גבוה כרצפים שכן נקשרו. לאחר מכן ערבבנו את הקבצים עם seed קבוע שיאפשר שחזור זהה בכל הרצה ולבסוף ייצגנו כל רצף באמצעות onehot encoding. המוצא של שלב זה הינו שני וקטורים X ו Y כאשר X הינו הייצוג של הרצף ו Y הוא הלייבל.

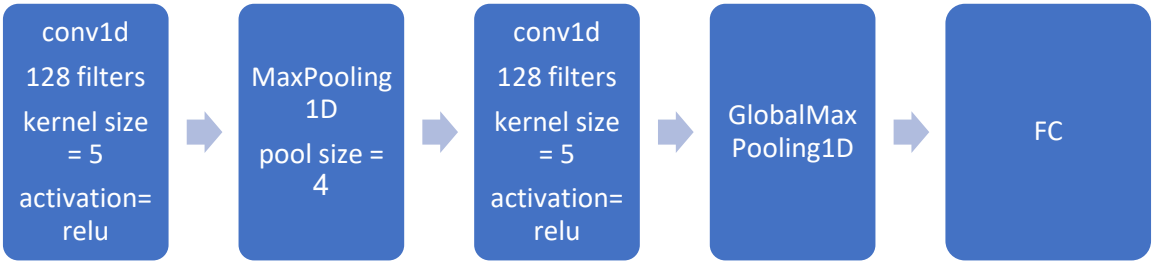




Example of X

שלב 2- ארכיטקטורת הרשת :

כאמור הרשת שלנו באה לפתור בעיית סיווג בינארי . הארכיטקטורה שבחרנו לעבוד איתה הינה :



כאשר ה FC (Fully Connected) הינו :

האופטימייזר אשר אנו עובדים איתו הינו Adam כאשר צעד הלמידה הנבחר הוא $Learning\ rate = 0.0005$

פונקציית השגיאה הינה כמובן $binary\ crossentropy$ כי אנו פותרים בעיית סיווג בינארית, בנוסף בחרתי לעבוד עם שתי מטריקות – AUC & Accuracy.

אחרי שבנינו את הרשת נאמן אותה בעזרת X ו Y אשר יצרנו בשלב העיבוד המקדים. תוך כדי חלוקת הנתונים לשני סטים – אחד לאימון ואחד לוולידציה. סט הוולידציה יכיל שליש מהדגימות וסט האימון שני שליש (גם פה ה seed קבוע למען שחזור התוצאות). אנו מקמפלים ומאמנים את המודל 5 פעמים לכל חלבון כאשר השוני בין הרצה להרצה הוא אתחול הפרמטרים ומשם אנו לוקחים את המודל בעל הדיוק הכי טוב.(מדד accuracy) בדומה למה שעשו deep bind בשלב 2 אצלם (אחרי שבחרו את הפרמטרים הכי טובים)

אציין שהמודל שלי קבוע מבחינת פרמטרים כפי שתיארתי למעלה: ניסיתי אינספור מודלים שונים כולל deep bind עם חיפוש היפר פרמטרים רחב ובכל התוצאות קיבלתי שלא משנה איזה מודל אני מריץ ואילו היפר פרמטרים הוגרלו התוצאות נשארות זהות פחות או יותר ואפילו אם הצלחתי לשפר את מדד ה AUC על סט הוולידציה אז השיפור ממש זניח ולא בהכרח מעיד על שיפור של מדד פירסון על סט הטסט של רצפי ה RNCMPT. בנוסף אציין שבכל המודלים שבניתי השיפור המשמעותי בחיזוי סט הוולידציה קורה ב epoch הראשון ולאחריו לרוב אין שיפור

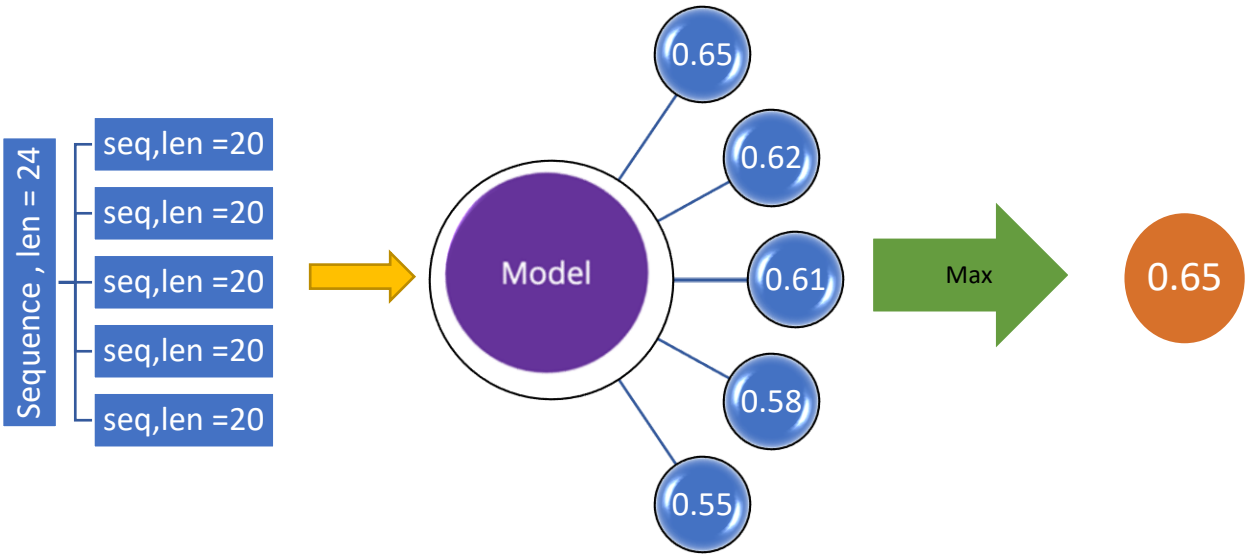
וגם אם יש שיפור אז הוא זניח וריצה של יותר epochs לרוב מורידה את הקורלציה עם סט הטסט ופוגעת במדד פירסון. מכאן הפרמטרים איתם בחרתי לרוץ בפקודת ה fit הינם: $batch\ size = 256, epochs = 1$.

שלב 3 – עיבוד סופי:

בשלב זה לקחנו את הרשת המאומנת וביצענו חיזוי על סט הטסט – RNCMPT.

רצפי ה RNCMPT מכילים רצפים באורכים שונים וצריך להתאים אותם למבנה הכניסה של הרשת שלנו. הפתרון שאנחנו עשינו הוא לחלק כל רצף לרצפים חופפים באורך 20 ככה שעבור רצף מסוים באורך len נקבל $len - 20 + 1$ רצפים חופפים. ככה אנו מחלקים את כל הסט לרצפים חופפים בגודל 20 ומבצעים predict עם המודל שאימנו. לאחר שקיבלנו את התוצאות – אשר יהוו הסתברות קשירה לאותו חלבון מסוים עבור הרצפים האלו. אנו לוקחים לכל רצף את התוצאה המקסימלית מבין כל הרצפים החופפים ששייכים לו. כלומר, לכל רצף באורך len יש $len - 20 + 1$ רצפים חופפים, לכל אחד מאותם רצפים השייכים לו יש את ההסתברות שהרשת חזתה ואנו לוקחים את המקסימלי.

כמובן שמספר הרצפים החופפים משתנה מרצף לרצף בסט הטסט והוא תלוי בגודל הרצף אותו אנו מחלקים.



המחשה – לקחנו רצף באורך 24,פירקנו ל 5 רצפים חופפים באורך 20 ומשם לכל רצף חישבנו הסתברות ולקחנו את המקסימום מבין 5 ההסתברויות שיצאו.(אם הרצף היה בגודל 30 ולא 24 אז התהליך היה זהה פשוט היו מתקבלים 11 רצפים חופפים ו 11 הסתברויות וגם שם המקסימום היה מבין אותם 11).

תוצאות :

RBNS	Pearson	time (sec)
RBP1	0.451916	139.526413
RBP2	0.410722	138.459044
RBP3	0.049687	137.745068
RBP4	0.194354	137.109782
RBP5	0.26739	139.648214
RBP6	0.431767	136.968045
RBP7	0.420717	139.062973
RBP8	0.131493	136.85477
RBP9	0.167082	143.02859
RBP10	0.110479	137.287058
RBP11	0.167261	144.746676
RBP12	0.42577	136.835078
RBP13	0.497872	138.152413
RBP14	0.078692	136.427236
RBP15	0.196136	148.875282
RBP16	0.483407	137.624792
Average	0.280297	139.271965

כפי שניתן לראות הגענו לממוצע פירסון 0.28. זמן הריצה לכל חלבון זניח (2 דקות) כי אנו רצים על מודל קבוע ללא חיפוש היפר פרמטרים.

```
model.summary()

Model: "sequential_36"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d_59 (Conv1D)           (None, 16, 128)           2560
_________________________________________________________________
max_pooling1d_27 (MaxPooling (None, 4, 128)            0
_________________________________________________________________
dropout_96 (Dropout)         (None, 4, 128)            0
_________________________________________________________________
conv1d_60 (Conv1D)           (None, 4, 128)            49152
_________________________________________________________________
global_max_pooling1d_30 (Glo (None, 128)               0
_________________________________________________________________
flatten_34 (Flatten)         (None, 128)               0
_________________________________________________________________
dense_137 (Dense)            (None, 64)                8256
_________________________________________________________________
dropout_97 (Dropout)         (None, 64)                0
_________________________________________________________________
dense_138 (Dense)            (None, 32)                2080
_________________________________________________________________
dropout_98 (Dropout)         (None, 32)                0
_________________________________________________________________
dense_139 (Dense)            (None, 32)                1056
_________________________________________________________________
dense_140 (Dense)            (None, 1)                 33
=================================================================
Total params: 63,137
Trainable params: 63,137
Non-trainable params: 0
```

מבחינת משקולות המודל שלנו מכיל כמעט 64K משקולות כפי שניתן לראות.