Markov Networks as Prior Distribution for Multi-Label Classification (in Wikipedia)

#### By: Alon Wolf (318336930) and Aviv Hadar (pi)

# תקציר - Abstract *// בסוף*

# תאוריה

* מודל מרקובי pairwise *// אביב*
* אלגוריתם צ'או ליו *// אביב*
* שימוש אלגוריתם צ'או ליו כ-prior *// אביב*
* "הכללה" לקורלציות שליליות *// אביב*

כפי שציינו קודם, במאמר נעסוק בשיפור ביצועי אלגוריתם בבעיית multilabel בעזרת יצירת prior על ה-labels בעזרת אלגוריתם "Chow Liu" ומודל מרקובי מעל להתפלגות הלייבילים.

לכל דאטאסט multilabel עם לייבלים:

עבור מודל הסתברותי-Pairwise Markov Network על הלייבלים שלו, מתקיים שהנוסחה ל-Inference זו:

ואם נניח שמבנה הרשת הוא עץ נקבל שהמשקלים האופטימליים לרשת הם:

כאשר מתקיים:

לכל classifier נזכור שמעניין אותנו חישוב ה-Marginals כדי לדעת את הסבירות של כל לייבל להיות חיובי. ניתן לחשב Marginals של מודל מרקובי בעזרת שיטת ה-Belief Propagation בצורה הבאה:

כאשר מתקיים:

משוואה (3) בעצם נותנת לנו את ההסתברות לקיומו של לייבל מסוים לפי **התפלגות הלייבלים בלבד**, זאת מפני שהמודל הנ"ל ממשקל את כל ערכי באופן זהה ו- מושפע מ- בלבד (בהינתן הצבת המשקלים האופטימליים). כלומר המודל מתעלם כרגע מהסבירות לערכי לייבל שונים, שזו יכולה באופן משמעותי להשתנות בין דוגמאות שונות בהתאם לערכי הפיצ'רים.

המטרה שלנו בפרויקט הייתה לבצע fine tuning לביצועי מודל ע"י prior שמשתמש בקונספט של רשתות מרקוביות כדי למדל את התפלגות הלייבילים, הוספנו לנוסחת חישוב ה-Marginal משקול ערכי בכל הודעה לפי .

כאשר מתקיים:

כלומר לשלב את הסבירות לקיומו של לייבל בהינתן הפיצ'רים () עם הסבירות שנובעת מהתפלגות הלייבלים (), במטרה לשפר את ביצועי המודל.

*הנוסחה המשולבת היא להלן:*

*חשוב לציין ש- אינה פונקציית הסתברות שכן נדרש לנרמל אותה:*

*זו הפרדיקציה לכל לייבל שמוחזרת מהפריור.*

## אינטואיציה – חשבתי להוסיף איזה משפט שניים על אינטואיציה ללמה זה הגיוני.

*אך עדיין לא ציינתי כיצד בחרנו את מבנה העץ של המודל המרקובי. תחילה עשינו זאת ע"י אלגוריתם Chow Liu באופן הבא:*

1. *הגדרנו גרף קשרים בין הלייבלים עם המשקלים הבאים:*
2. *לאחר מכן מצאנו עץ פורש מקסימאלי לגרף*
3. *והעץ שקיבלנו שימש אותנו לבניית המודל המרקובי*

*בהמשך בחנו את המודל תחת מבני עץ שנבנו בצורות שונות מזו, יוצג בחלק הבא.*

*כעת נתאר כיצד בחנו את השפעת המודל הנ"ל על ביצועי אלגוריתם ML.*

## Partial MI Chow Liu

# דאטא // אלון

את המידע חילצנו מאתר ויקיפדיה. השתמשנו בספריות wikipediaapi, wptools על מנת לחלץ את הטקסט וה-infobox של ערכי הויקיפדיה (דוגמה לערך ניתן לראות ב*איור 1*).

נשאלת השאלה אילו ערכים לקחת – יכלנו לעשות רשימה של ערכים ולבחור מתוכם אך על מנת שההתפלגות של הערכים תהיה זהה (שהערכים ייבחרו על ידי התפלגות אחת) לקחנו את כלל הערכים מתוך קטגוריה אחת: מוזיקאים (<https://en.wikipedia.org/wiki/Category:Musicians>).

לקחת את הטקסט כמשתנה הבלתי-תלוי ואת השאלות של ה-infobox כמשתנה התלוי.

את הטקסט העברנו טוקניזציה פר מילה כך שלכל מילה ייצרנו מימד חדש על האם הטקסט הכיל את המילה. לכל מילה הכנסנו ציון לפי אלגוריתם TF-IDF כך שככל שמילה מופיע יותר בטקסט כך היא מקבלת ציון גבוה יותר, מנורמל לפי מס' הפעמים שהמילה הופיעה בכלל הטקסטים. הורדנו מילים שחזרו במס' בודד של טקסטים כיוון שלא רצינו שהמודל ילמד קורלציה שרק קורלטיביות למילים בודדות.

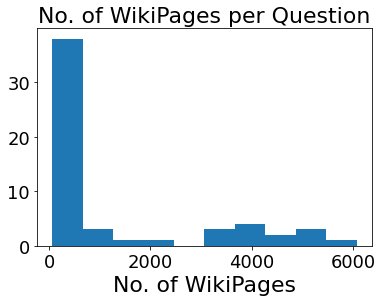
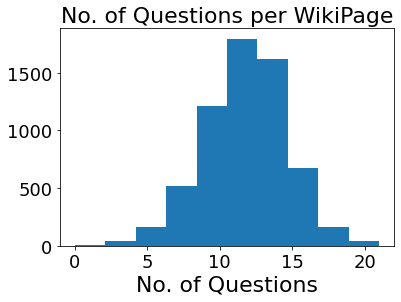
כמו כן, מתוך השאלות גם כן סיננו שאלות לא נפוצות (מופיעות עבור מס' בודד של ערכים) כיוון שעבור דוגמאות כאלו הציונים הסופיים מאוד משתנים לפי החלוקה של קבוצת האימון וקבוצת המבחן ולכן מאוד לא יציבים.

כפי שניתן לראות מהדוג', הבעיה הינה MultiLabel כיוון שלכל ערך ישנו מס' משתנה של שאלות.

את התפלגות מס' השאלות לכל ערך ניתן לראות באיור מס' 2.

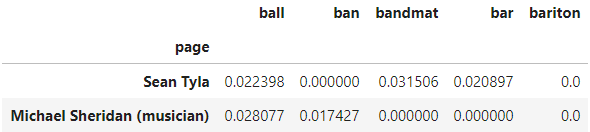


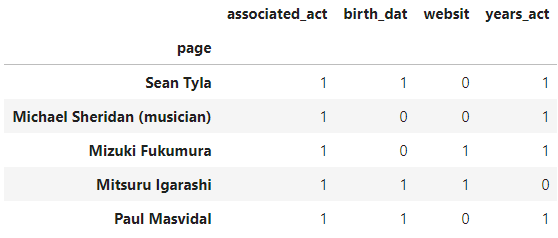
*איור 1 – ערך ויקיפדיה של אלטון ג'ון. באדום – הערך (טקסט), בכחול – עובדות מהירות. מטרת הפרויקט היא לחזות את החלק הכחול בהינתן האדום.*



## סה"כ

קיבלנו 6221 דוגמאות עם 4328 פיצ'רים ו-56 סוגי מחלקות.





# תהליך // אלון

התחלנו בלייצר מודל לכל Class. השתמשנו במודל OneVsRest – כלומר, יצרנו מודל לכל מחלקה בנפרד. עשינו זאת כיוון שזהו common practice שמשתמשים בו כשרוצים למדל בעיות של multilabel (כיוון שאיננו יכולים להשתמש במודל שיחזיר מחלקה אחת לכל תצפית כיוון שלא כך). נשים לב שהמודל לא מתחשב בקשרים בין המחלקות כיוון שהוא מנתח כל מחלקה בנפרד.

על כן, רצינו להוסיף "prior" אשר כן יתחשב בקשרים בין המחלקות כך שאם יש קורלציה (MI) בין 2 מחלקות אז זה ישפיע על הציונים שהוא נותן. רצינו לדעת הם וכמה אנחנו יכולים לשפר את המודל הראשוני תוך שימוש ב"prior" על הדוגמאות. את ה-prior מימשנו באמצעות מודל מרקובי בין המחלקות.

## Prior (p\_ij)

עבור כל מחלקה i,j יצרנו את הסיכוי ש-i,j מופיעים ביחד:

כאשר N הוא מספר הדוגמאות.

בעזרת הקירוב לפרדיקציה יצרנו קירוב ל-MI בין 2 מחלקות:

כעת, יצרנו עץ פורש מקסימלי. בשביל לשערך את המשקלים השתמשנו ב-Belief Propagation.

יצרנו פרדיקציה עבור כל תצפית וכן עבור כל מחלקה (נסמן ב- את הפרדיקציה עבור הדוגמא ה-i למחלקה c – ה-proba). ככתוב בחלק התאורטי.

# תוצאות // אלון

## אבלואציה של המודל

מדדנו את ה-ROC-AUC וה-F\_1 של הפרדיקציות לפני ואחרי ה-prior. כיוון ש-F1 דורש *להקשיח את הפרדיקציות בחרנו את ה-th אשר ממקסם את F1 על ה-validation.*

*מדדנו את הדלתא של התוצאות עבור מודלי בסיס שונים.*

*תוצאות:*

* לכל מטריקה [roc, f1]
  + לכל מודל
    - לכל prior
      * מה התוצאות
* הסברים (+ גרפים) *//* נכתוב ביחד

# דברים נוספים שניסינו

# סיכום ומסקנות // בסוף