קורס: קודים לתיקון שגיאות ושימושיהם במדעי המחשב
מרצה: ד"ר קלים יפרמנקו
סמסטר סתיו תשפ"א
08/12/2020

הרצאה 16:

פענוח רשימה: List-decoding

מבוא:

עד כה נחשפנו למודל אחד של פענוח, פענוח יחיד (Unique Decoding) בו משוחזרת מילת קוד אחת אשר חייבת להיות מילת הקוד המקורית. בתורת הקידוד, פענוח רשימה (List-decoding) הוא חלופה לפענוח יחיד (Unique -decoding) של קודי תיקון שגיאות עם שיעורי שגיאה גדולים.

המושג הוצג לראשונה על ידי פיטר אליאס ב-1957 . הרעיון העיקרי מאחורי פענוח רשימה הוא שאלגוריתם הפענוח מוציא רשימת הודעות אפשרויות שאחת מהן נכונה (במקום להוציא הודעה אפשרית אחת). פענוח רשימה מאפשר פענוח גם בהינתן מספר רב יותר של שגיאות (מאשר זה שמאפשר פענוח יחיד), שיכול להגיע כמעט עד למרחק הקוד ממש .

תזכורת: בהינתן שהמרחק של הקוד הוא d. קוד יכול לתקן, כלומר לפענח: $\left\lfloor\frac{d-1}{2}\right\rfloor$ שגיאות.

ישנם שני מודלים עיקריים למידול הרעש:

1) מודל רעש הסתברותי(נחקר על ידי Shannon) - בו רעש הערוץ ממודל, כלומר ההתנהגות ההסתברותית של הערוץ ידועה וההסתברות להתרחשות שגיאות רבות מדי או מעטות היא נמוכה. במודל זה יש שגיאות אקראיות (*random errors*) אשר נפתרות באופן מקרי ואז לרוב ה-*pattern* של השגיאות מפוענח בצורה נכונה.

2) מודל רעש היריב (שהוצג על ידי Hamming) - בו הערוץ פועל כיריב שמשחית באופן שרירותי את מילת הקוד בכפוף למספר השגיאות הכולל. במודל זה יש שגיאות אדורסריאליות (*adversarial errors*), שגיאות מסוג זה נגמרות כך: היריב רואה את הקוד ואת אלגוריתם הפענוח ואז הוא "בוחר" את השגיאות.

ההבדל המהותי ביניהם: קיים פער בין ביצועי תיקון השגיאות במודלים אלו- קודים המבוססים על המודל הראשון, הם הטובים ביותר, בהם הבנייה אקראית ומאפשרת לנו לבנות קודים כאשר: $R = 1 - H(\frac{d}{n})$, וכך ניתן לפענח מ-d שגיאות, אבל כאשר השגיאות הן אקראיות אז אפשר לפענח מ- $\frac{d}{2}$ שגיאות. כלומר, קיים פער של פקטור 2 בין שגיאות אקראיות לשגיאות אדורסריאליות.

איך נגשר על הפער? על מנת לגשר על הפער נלמד על פענוח רשימה. פענוח זה הביא לכך שגם עבור המודל השני, ניתן להשיג את התמורה האופטימלית- בין קצב העברת המידע לכמות השגיאות שניתן לתקן. במובן מסוים, פענוח זה משפר את ביצועי תיקון השגיאה להיות כמו במודל רעש הראשון.

באופן לא פורמלי- נאמר שקוד ניתן לפענוח רשימה (List Decodable) אם בהינתן מילת קוד מורעשת, ניתן לשחזר ממנה רשימה קצרה של מילות קוד אפשריות כך שמובטח שאחת מהן היא מילת הקוד המקורית. כלומר, אלגוריתם הפענוח שלנו יחזיר רשימה קצרה של תשובות, כך שמובטח לנו (במצב בו לא קיבלנו יותר מדיי תשובות) שאחת מהתשובות הללו היא התשובה הנכונה, אך לא נדע מהי התשובה הנכונה מתוך הרשימה.

תיאור התהליך:

אליס רוצה לשלוח הודעה m לבוב, אז ראשית היא מקודדת את ההודעה c(m) ולאחר מכן היא שולחת את הקידוד הנ"ל לבוב. קידוד זה מגיע אליו בתוספת רעש: $c(m) + e$. בוב מפעיל את אלגוריתם הפענוח על ההודעה שקיבל ומחזיר רשימה כך שאם אורכה קצר אז התשובה הנכונה תהיה חלק מהרשימה הנ"ל:

$$D(c(m) + e) = [m_1, m_2, \dots, m_i]$$

למה ישנה חשיבות לכך שהרשימה תהיה קצרה?

באופן עקרוני, נוכל תמיד להחזיר רשימה שהיא אוסף כל ההודעות וכך יובטח שאחת מהן היא הנכונה, אך במקרה בו אורך הרשימה הוא אקספוננציאלי, החזרה של כל הרשימה היא מאוד לא יעילה, ולכן נרצה שאורך הרשימה שיוחזר יהיה קצר- כלומר, יהיה קבוע או פולינומי באורך ההודעה: $poly(\mathrm{k})$.

האם נעדיף קודים שהם List-decodable או קודים שהם Unique- decodable?

Unique- decoding נותן תשובה יחידה ונכונה ו-List-decoding נותן רשימה שבה נמצאת התשובה הנכונה, אך צריך להבין מהי. כלומר יש איזשהו חוסר וודאות מה תהיה התשובה הנכונה, זה בעצם החסרון של List-decoding ולכן באופן עקרוני קודים שהם Unique- decoding הם עדיפים.

מהו היתרון של List-decoding?

במקרה בו אנו מקבלים R, δ טובים יותר, אז פה נוכל להתמודד עם מספר רב יותר של שגיאות, באופן לא פורמלי, נוכל להתמודד עם פי 2 שגיאות. מבחינת הפרמטרים List-decoding מקבל את הפרמטרים של שגיאות אקראיות, והוא יכול להתמודד עם שגיאות שהן שגיאות אדברסליות.

דוגמאות לשימוש ב- List-decoding:

1. מקרה בו שולחים הודעה שהינה מפתח לאיזשהו צופן, אז אלגוריתם הפענוח יחזיר רשימה של מפתחות. נוכל לעבור על כל אחד מהמפתחות ולבדוק אם הוא המפתח לצופן או לא, במצב הנ"ל האלגוריתם מבטיח לנו שהמפתח הנכון יהיה אחד האיברים ברשימה שתתקבל.

2. תקשורת אינטראקטיבית- סטודנט שואל שאלה בשיעור, אך המרצה לא שומע אותו טוב. אז הסטודנט לוקח את השאלה שלו ומקודד אותה עם קוד שהוא List-decodable ושולח אותה למרצה. המרצה מקבל רשימה של שאלות שאחת מהן היא השאלה שהסטודנט באמת שאל, כלומר המרצה יכול לענות על כל השאלות ולהחזיר את התשובות הנ"ל לסטודנט, וכך הסטודנט יבחר את התשובה הרלוונטית עבורו וישמע אותה. במקרה הזה, המרצה מחזיר את התשובות לכל השאלות, נשים לב שפעולה זו אומנם מגדילה מעט את ה-Rate אך זה עדין מאוד שימושי.
הערה: כאשר בונים פרוטוקול תקשורת מתקשורת אינטראקטיבית אז List-decoding זה כלי מאוד שימושי ובלעדיו אי אפשר להשיג תוצאות אופטימליות (אפילו בשאלות של Unique-decoding).

3. מקרה בו נקבל הודעה אשר השגיאות בה הן אקראיות, אז בסיכוי גבוהה אורך הרשימה הוא 1.

הגדרה פורמלית- פענוח רשימה (List-decoding):

עבור $0 \le \rho \le 1$ ו- $L \ge 1$ קוד $C \subseteq \sum^n$ נקרא List-decodable אם לכל $y \in \sum^n$ (כלומר, לכל הודעה שהתקבלה) מתקיים:

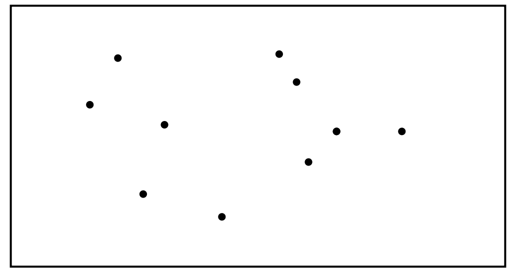
$$|\{c \in C \mid \Delta(y, c) \le \rho \cdot n\}| \le L$$

הערה: קוד c נקרא: $(\rho \cdot n, L) -$ List decodable

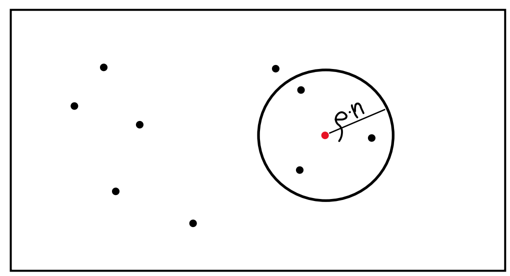
קורס : קודים לתיקון שגיאות ושימושיהם במדעי המחשב
מרצה : ד״ר קלים יפרמנקו
סמסטר סתיו תשפ״א
08/12/2020

<u>הסבר של ההגדרה באמצעות דוגמה :</u>

נניח שקיים קוד c כזה :



מה זה אומר שהקוד הוא $(\rho \cdot n, 3) -$ List decodable ?



עבור <u>כל</u> נקודה שנבחר (באיור זו הנקודה האדומה), אם ניצור סביבה כדור ברדיוס $\rho \cdot n$ אז מובטח לנו שמספר הנקודות של הקוד שיהיו בתוך הכדור הנ״ל הוא לכל היותר L , במקרה שלנו לכל היותר 3.

<u>על מנת להוכיח שהאלגוריתם פענוח רשימה עובד, נצטרך להראות שני דברים :</u>

1. אורך הרשימה שהוחזרה קצר.
2. מילת הקוד הנכונה, כלומר המקורית שנשלחה נמצאת ברשימה הנ״ל.

<u>למה ההודעה המקורית נמצאת ברשימה הקצרה שהוחזרה?</u>

אליס רוצה לשלוח הודעה m לבוב, אז ראשית היא מקודדת את ההודעה c(m) ולאחר מכן היא שולחת את הקידוד הנ״ל לבוב. קידוד זה מגיע אליו בתוספת רעש. כאשר $wt(e) \ \leq \rho \cdot n$, כלומר מספר השגיאות שהתווספו הוא לכל היותר $\rho \cdot n$, בוב מקבל $y = c(m) + e$ ומסתכל על כל מילות הקוד c שמקיימות : $\Delta(c, y) \leq \rho \cdot n$ ואותן הוא מוסיף לרשימה. מכיוון שמובטח לנו שאורך הרשימה שבוב יחזיר הוא לכל היותר L , אז נקבל שהמרחק בין ההודעה שבוב קיבל לבין ההודעה הנכונה הוא : $d(\Delta(c(m), y)) \leq \rho \cdot n$ ולכן, קיבלנו בעצם שמילת הקוד המקורית $c(m)$ תמצא בוודאות ברשימה שהוחזרה.

<u>מה יהיה המרחק של הקוד כאשר נתון שהקוד הוא $(\rho \cdot n, 1) -$ List decodable, כלומר $L = 1$?</u>

באותו אופן, עבור <u>כל</u> נקודה שנבחר אם ניצור סביבה כדור ברדיוס $\rho \cdot n$, מובטח לנו שבתוך הכדור הנ"ל תהיה נקודה אחת לכל היותר. מכיוון שהמרחק של הקוד הוא המרחק בין שתי נקודות, אז המרחק במקרה הזה הוא הקוטר של המעגל, כלומר: $2 \cdot \rho \cdot n$

<u>מסקנה 1-</u> אם מספר השגיאות הוא לכל היותר $\rho \cdot n$ אז אנחנו יכולים להוציא רשימה באורך אחד.

נוכל לפענח Unique-decodable מ- -$\frac{d}{2}$ שגיאות, והמרחק של הקוד הוא פי 2 מזה אז לקוד יש מרחק: $2 \cdot \rho \cdot n$

<u>מסקנה 2-</u> עבור כל קוד, אם $d(c) = d$ אז הקוד הוא: $(\left\lfloor\frac{\mathrm{d}-1}{2}\right\rfloor, 1) - \mathrm{List} - \mathrm{decodable}$

אנו יודעים לגבי קודים רגילים עם מרחק d – שעבור <u>כל</u> נקודה שנבחר אם ניצור סביבה כדור ברדיוס $\frac{d-1}{2}$, אז בתוך הכדור הנ"ל תהיה נקודה אחת של הקוד לכל היותר, מכיוון שאם יהיו שתי נקודות אז המרחק ביניהן צריך להיות קטן מ-d.

הערה: באופן עקרוני, ככל ש-L יותר קטן זה יותר טוב וככל ש- ρ יותר גדול זה יותר טוב.

<u>נחתור לאיזשהי הכללה ל*Gillbert-Varshamov* עבור איזה R, ρ יש קודים שהם List decodable?</u>

כל עוד האורך של הרשימה שלנו הוא לא אקספוננציאלי אז הוא לא משפיע על הקצב של הקוד ולא על מספר השגיאות שהקוד שלנו יכול להתמודד כלומר אורך הרשימה לא מאוד משפיע על ה- R, ρ.

<u>List-decoding capacity:</u>

נניח שקיים קוד c ונתון שקצב העברת המידע הוא: $R = \frac{k}{n}$. מה מספר השגיאות המקסימלי שנוכל לפענח?
כלומר מה ה- ρ המקסימלי כך שהקוד c הוא: $(\rho \cdot n, poly(\mathrm{k})) -$ List decodable?

<u>נענה על זה באמצעות המשפט הבא:</u>

<u>משפט:</u> עבור $q = 2$ ו- $0 \leq \rho \leq \frac{1}{2}$ ולכל $\varepsilon > 0$, קיים N כך שלכל $n > N$ מתקיים:

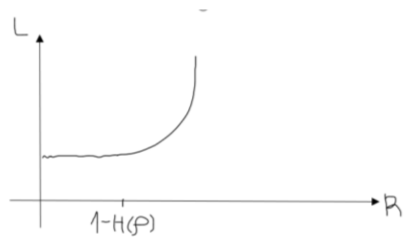
1. אם $R < 1 - H(\rho) - \varepsilon$ אז קיים קוד שהוא: $(\rho \cdot n, O(\frac{1}{\varepsilon})) -$ List decodable

הערה: זה אומר שהיחס בין ה-*Rate* של הקוד למספר השגיאות שהוא יכול להתמודד איתן, הוא כמו היחס בין ה-*Rate* של הקוד למספר השגיאות שהוא יכול להתמודד איתן, במקרה של רעש אקראי.

2. אם $R > 1 - H(\rho) + \varepsilon$ אז לכל קוד שהוא $(\rho, L) -$ List decodable מתקיים: $L \geq 2^{\Omega(n)}$

<u>משמעות המשפט:</u>

משפט זה אומר שעבור $Rate$ שמתקרב ל-capacity, אך עדין מתחתיו, נוכל לקבל ρ שהוא לא n עם אורך רשימה שהוא פחות או יותר קבוע $(\frac{1}{\varepsilon})$, כלומר לכל היותר באורך פולינומי. דבר המאפשר אלגוריתמי פענוח יעילים. ואילו עבור $Rate$ שעולה על ה-capacity, נקבל שאורך הרשימה הופך לאקספוננציאלי, דבר הפוסל את קיומם של אלגוריתמי פענוח יעילים.



<u>הוכחת המשפט:</u>

הערה: הוכחה זו דומה להוכחה של משפט Shannon לקיבולת של הערוץ הסימטרי הבינארי BSC(p).

<u>הוכחה של (1):</u>

נניח ש- $R < 1 - H(\rho) - \varepsilon$ וצ״ל שקיים קוד שהוא: $(\rho \cdot n , O(\frac{1}{\varepsilon})) - \text{List} - \text{decodable}$:

נבחר קוד מקרי, ונוכיח שההסתברות שהקוד הזה הוא $(\rho \cdot n, L) - \text{List} - \text{decodable}$ היא חיובית, כלומר בסיכוי טוב. אז זה אומר בפרט, שקיים קוד שהוא $(\rho \cdot n, L) - \text{List} - \text{decodable}$

<u>איך נוכיח את זה?</u>

נבחר את גודל הקוד c להיות: $|c| = 2^{k}$ כאשר כל מילת קוד נבחרת בצורה אחידה מכל מילות הקוד ובאופן בלתי תלוי.

<u>מה צריך לקרות כדי ש-c לא יהיה List – decodable? כלומר לא יהיה $(\rho \cdot n, c) - \text{List} - \text{decodable}$?</u>

מהגדרת $(\rho \cdot n, c) - \text{List} - \text{decodable}$:

$$|\{c \in C \mid \Delta(y,c) \leq \rho \cdot n\}| \leq L$$

צריך להיות איזשהו y כך שבכדור שמסביבו ברדיוס $\rho \cdot n$ מספר מילות הקוד יהיו: $L + 1$.

במילים אחרות קיים y כך ש:

$$|\{c \in C \mid \Delta(y,c) \leq \rho \cdot n\}| \geq L + 1$$

דרך נוספת לתאר את זה: $|B(\rho \cdot n, y) \cap c| \geq L + 1$

הערה: $B(\rho \cdot n, y)$ - סימון לכדור ברדיוס $\rho \cdot n$ מסביב ל-y .

נמצא חסם עבור ההסתברות שהקוד הוא לא List – decodable :

עבור y ספציפי ועבור מילת קוד ספציפית $c_i \in \{c_1, c_2, \dots, c_{2^k}\}$, נחשב את ההסתברות שמילת קוד c_i שייכת לכדור היחידה ברדיוס של $\rho \cdot n$:

$$P_r(c_i \in B(\rho \cdot n, y)) = \frac{|B(\rho \cdot n, y)|}{2^n} = \frac{2^{H(\rho)\cdot n}}{2^n} = 2^{-n(1-H(\rho))}$$

הסבר: מעבר ראשון – יש 2^n אפשרויות לבחור את c_i ומתוכן גודל הכדור, משמע נפח הכדור שרדיוסו $\rho \cdot n$ ומרכזו היא המילה שהתקבלה- y . המעבר השני מתבסס על חישוב שראינו בעבר שהוא נכון עד כדי אפסילון.

תזכורת: שני מאורעות הם בלתי תלויים אם ההסתברות ששניהם יקרו שווה למכפלת ההסתברויות של כל מאורע בנפרד, כלומר: $P_r(A_1 \cap A_2) = P_r(A_1) \cdot P_r(A_2)$

כעת נחשב מה ההסתברות ש- $L+1$ מילות הקוד הראשונות $(c_1, c_2, \dots, c_{L+1})$ יהיו כולן בתוך הכדור $B(\rho \cdot n, y)$. כלומר, במצב זה הקוד שלנו הוא לא List – decodable

בחירת מילות הקוד נעשית באופן בלתי תלוי ולכן:

$$P_r\left(c_1 \in B(\rho \cdot n, y) \wedge \dots \wedge c_{L+1} \in B(\rho \cdot n, y)\right) = P_r(c_1 \in B(\rho \cdot n, y)) \cdot \dots \cdot P_r(c_{L+1} \in B(\rho \cdot n, y))$$

$$= \prod_{i=1}^{L+1} P_r(c_i \in B(\rho \cdot n, y)) = 2^{(L+1))} 2^{-n(1-H(\rho))} = 2^{-n(1-H(\rho))\cdot(L+1)}$$

בעצם הראנו שעבור c ספציפי, הסיכוי שהוא יהיה בכדור מסביב ל-y הוא קטן.

תזכורת: Union Bound אומר ש: $P_r(A_1 \cup A_2) \le P_r(A_1) + P_r(A_2)$

נחשב את ההסתברות שהקוד המיקרי שבחרנו הוא לא List – decodable:

$$P_r\left((\rho \cdot n, L) - List - decodable \text{ הוא לא } c - \text{ש}\right) =$$

$$P_r(\exists y \in \{0,1\}^n, \exists (i_1, \dots, i_{L+1} \in 2^k) | c_{i1} \in B(y, \rho \cdot n), \dots, c_{iL+1} \in B(y, \rho \cdot n)) \le$$

$$\sum_{y \in \{0,1\}^n} \sum_{i_1, \dots, i_{L+1} \in [2^k]} P_r\left(c_{i1} \in B(y, \rho \cdot n), \dots c_{iL+1} \in B(y, \rho \cdot n)\right) \le * \; 2^n \cdot 2^{(L+1)k} \cdot 2^{-n(1-H(\rho))\cdot(L+1)}$$

*מס׳ האפשרויות לבחירת y כפול מס׳ האפשרויות לבחירת i כפול החסם שחישבנו) ונבצע מעבר אלגברי:

$$= 2^{n+(L+1)k-n(1-H(\rho))\cdot(L+1)} =* \; 2^{-n\cdot(L+1)(1-H(\rho)-R-\frac{1}{L+1})} =* \; 2^{-n\cdot(L+1)(1-H(\rho)-R-\frac{\varepsilon}{2})} \le$$

$$\le \; * \; 2^{-n\cdot(L+1)\cdot\frac{\varepsilon}{2}} < 1$$

*נציב: $k = R \cdot n$ ונוציא גורם משותף. *נבחר: $\frac{2}{\varepsilon} = L + 1$ ונציב. *נציב: $R < 1 - H(\rho) - \varepsilon$

<u>לסיכום:</u> הראנו שההסתברות הנ"ל קטנה מ- 1, ולכן המאורע המשלים בהכרח חיובי, כלומר שההסתברות שהקוד הזה הוא $(\rho \cdot n, L) - \text{List} - \text{decodable}$ היא חיובית ולכן בעצם קיים קוד כזה.

<u>הוכחה של (2):</u>

נניח ש- $R > 1 - H(\rho) + \varepsilon$ צ"ל: לכל קוד שהוא $(\rho, L) - \text{List decodable}$ מתקיים: $L \geq 2^{\Omega(n)}$:

נבחר את y להיות משתנה מקרי, שהוא מספר הנקודות $c \in C$ שהן בתוך הכדור, כלומר:

$$z(y) = |\{c \in C | c \in B(\rho \cdot n, y)\}|$$

<u>נסמן:</u>

$$z_i = \begin{cases} 1 & c_i \in B(\rho \cdot n, y) \Leftrightarrow y \in B(\rho \cdot n, c_i) \\ 0 & c_i \notin B(\rho \cdot n, y) \end{cases}$$

אז $z(y) = \sum_{i=1}^{2^k} z_i(y)$.

<u>כעת נחשב את התוחלת של z, כלומר את $E(Z)$:</u>

$$E(Z) = \sum_{i=1}^{2^k} E(z_i) = \sum_{i=1}^{2^k} P_r\left(y \in B(\rho \cdot n, c_i)\right) = \sum_{i=1}^{2^k} 2^{-\mathrm{n}(1-H(\rho))} = 2^k \cdot 2^{-\mathrm{n}(1-H(\rho))} =$$

$$=_* 2^{-\mathrm{n}(1-H(\rho)-R)} \geq_{**} 2^{\varepsilon \cdot \mathrm{n}}$$

* נציב: $k = R \cdot n$

**נציב: $R > 1 - H(\rho) + \varepsilon$

<u>לסיכום:</u> בחרנו את y באופן מקרי וקיבלנו שתוחלת מילות הקוד בכדור ברדיוס $\rho \cdot n$ מסביב ל-y היא: $2^{\varepsilon \cdot \mathrm{n}}$ כלומר, קיים y כלשהו כך שבכדור ברדיוס $\rho \cdot n$ יש $2^{\varepsilon \cdot \mathrm{n}}$ מילות קוד, ולכן כאשר $L < 2^{\varepsilon \cdot \mathrm{n}}$ אז הקוד הוא לא $L - \text{List} - \text{decodable}$.