הערות למצגת בנושא *QA*

סמינר בעיבוד שפה טבעית, פרופ׳ נחום דרשוביץ

שקופית 1 : פתיחה

ההרצאה היום תהיה על תחום ה-*Question Answering*

שקופית 2 : *Overview*

- הנושאים בהם אגע היום :
- מה זה *Question Answering*?
- עקרונות כלליים ב-*QA*, גישות וטכניקות למציאת תשובות
- רקע על מערכות *QA* ותחומי עניין עיקריים
- *TREC – Text retrieval conferences* – מה זה ומה הקשר ל-*QA*
- נושא המאמר בו בחרתי שהוא מערכת *QA* מבוססת על גרפים סמנטיים

שקופית 3 : *What is question answering?*

- המטרה העיקרית : מערכות המסוגלות לספק תשובות לשאלות ע״י מציאת תשובות באוסף מסמכים (דפי אינטרנט או מסדי נתונים לוקאליים). בניגוד לתוצאות מנועי חיפוש, המטרה היא לתת תשובות מדוייקות ולא אוסף של מאות קישורים, שחלקם אינם עניינים לשאלה.
- כמות המידע ברשת עצומה וגדלה ללא הפסקה, כך שתהליך מציאת תשובות לשאלות נעשה קשה יותר ויותר, וטכנולוגיית ***QA*** נעשית יותר ויותר רלוונטית.
- כמובן שקשה להעריך ביצועי מערכות ***QA*** ודיוקן שכן לאנשים שונים יש תפיסות שונות בנוגע למה המרכיבים הנדרשים לתשובה אידיאלית, בייחוד עבור מערכות האמורות להחזיר תשובות מדוייקות.

שקופית 4 : *Approaching QA*

מחקר ה-*QA* מתמודד עם הצורך לתת מענה לטווח רחב של סוגי שאלות, למשל עובדות, הגדרות, איך/מי/מתי, היפותטיות וכו׳. תחום החיפוש משתנה :

- *Closed domain* : תחום חיפוש התשובה הוא דיסציפלינה או עולם ספציפי, למשל תחום הרפואה/מכונאות רכב/ביולוגיה. חיפוש בתחום סגור נראית קלה יותר שכן מערכות *QA* יכולות לנצל את מידע פרטני מסודר באונטולוגיות – ייצוג פורמלי של קונספטים בתחום מסויים והקשרים ביניהם – גרפים רעיוניים של אותו תחום.
- *Open domain* : חיפוש תשובה בכל נושא שהוא, כאשר תחום החיפוש הוא מידע ואונטולוגיות כלליות. מצד אחד תחום החיפוש רחב יותר ועל כן נראה קשה יותר למצוא תשובה ספציפית, אך מצד שני מהיותו תחום רחב לרוב יחזיק יותר מידע ממנו ניתן להוציא את התשובה. ה-*WWW* היא תחום פתוח, למשל.

ישנן גישות שונות לגשת למשימת חיפוש תשובה, אך ברוב המערכות משתמשים בתבנית הבאה :

- ניתוח וסיווג השאלה – בניסיון לקבוע איזה סוג של תשובה מחפשים. לרוב נעשה ע״י בחינת מילים המרכיבות את השאלה, למשל : *who is* מעיד שהתשובה הנדרשת היא שם כלשהו,*how many* מעיד שהתשובה הנדרשת היא מספר וכד׳.
- שימוש בטכנולוגיות הוצאת מידע לבניית מסד נתונים רלוונטי – טכנולוגיות דומות לאלו בהם נעשה שימוש במנועי חיפוש משמשים למציאת מסמכים רלוונטיים בהם סביר שתמצא התשובה הנדרשת. בד״כ תהליך זה נעשה ע״י חיפוש מונחים ומושגים מתוך השאלה.
- בשלב הבא נעשה ניתוח למסמכים הרלוונטים בהם נעשה חיפוש לישויות המתאימות לתבנית של התשובה המבוקשת. למשל, עבור השאלה מי המציא את הטלפון, ייעשה חיפוש לשמות. שני מושגים חשובים בתחום :
  - *Information extraction* : הוצאת מידע בנוי, מחולק לקטגוריות ומוגדר היטב מבחינת תוכן וסמנטיקה מתוך מסמכים לא מובנים. מטרה יותר ממוקדת היא ליצור מבנה לוגי כדי להוציא ממנו קשרים לוגיים.
  - *Information retrieval* : מדע העוסק בחיפוש מסמכים, חיפוש בתוך מסמכים, *IE* הוא תת תחום של *IR*.
- לבסוף נעשית בדיקת הישויות המועמדות, ונעשה שימוש במודולים להוצאת תשובה המחפשים רמזים נוספים כדי לאמת את נכונות התשובה, ואם נמצאה יישות מתאימה - היא תוחזר כתשובה.

שקופית 5 : *Where do answers come from?*

ישנן טכניקות *QA* פשוטות המשתמשות בשיטות מבוססות מילות מפתח כדי להוציא משפטים חשובים מתוך המסמכים שהוחזרו כפוטנציאלים להכיל תשובה. תשובות אפשריות מנותחות על בסיס תכונות סינטקטיות כגון סדר מילים ומיקומן או דמיון לשאלה. מערכות שונות, בעיקר אלו המסתמכות על

מאגר מסמכים גדול בו יתירות גבוהה, יכולות לחפש תשובה לפי תבניות מסויימות בהתאם לסוג וניסוח השאלה, לפעמים כך שהתשובה המבוקשת היא שינוי קל בצורת השאלה. למשל: עבור השאלה *"how many X is there in Y"* המערכת תחפש תשובה מהצורה *"there are Z X in Y"*. שיטה זו טובה בעיקר עבור שאלות מבוססות עובדה המחפשות שם, מקום, תאריך וכמות.

במקרים רבים שיטות מבוססות שינוי השאלה או מילות מפתח לא מספיקות ויש צורך בשיטות נוספות לעיבוד קונטקסט (הקשר), סינטקס וכמובן - **סמנטיקה**. שאלות *"why"* או *"how"* שאלות בעלות אילוצי זמן ומקום, שאלות בעלות ניסוח פגום או דו משמעות הן מקרים בהם יש צורך בשימוש בכלים מתקדמים יותר לניתוח השאלה וחיפוש התשובה, תוך שימוש בטכניקות עיבוד שפה טבעית שונות. מערכות *QA* מתקדמות ישתמשו לרוב גם במידע שלא מוצא רק מבנק המסמכים הנתון, אלא גם מאונטולוגיות כלליות כגון *WordNet*, שהיא כפי שכבר הוזכר בהרצאות קודמות מסד נתונים לקסיקלי לשפה האנגלית שמכיל קבוצות של מילים נרדפות הנקראות *synset* (המערכת שאציג משתמשת ב-*WordNet*), הגדרות וקשרים סמנטיים בין המילים הנרדפות. מאגר נתונים חשוב נוסף שכבר הוזכר הוא *VerbNet*, מאגר נתונים שמאגד מידע סמנטי וסינטקטי על פעלים והתפקידים הסמנטיים שלהם. להלן פירוט שיטות *Answer extraction* שונות:

- ***Word Matching***: ביצוע *Information retrieval* על טקסט היעד באמצעי *NLP* והחזרת משפטים אפשריים כתשובה מתוך הטקסט. ע"י התאמת מילים לשאלה ניתן דירוג למשפטים לפי כללים מסויימים של המערכת, וכך נקבעת התשובה.
- ***Question Classification***: קלאסיפיקציה של תשובות אפשריות לפי סוג סמנטי ולפי טקסונומיה (מיון) מסויים. למשל *Qtargets* הוא מאגר סוגים של שאלות שנבנה על בסיס מעל 17,000 שאלות. *Webclopedia* היא מערכת המשתמשת בסיווגים אלו כדי לצמצם את מרחב החיפוש לתשובות מסוג מסויים.
- ***Semantic parsing***:
  - ***Semantic type matching***: סוגים שונים של פריטי מידע, כגון מספרי טלפון, מיקוד ערים, כתובות דוא"ל וכמויות שונות מופיעים בתבניות לקסיקוגרפיות אופייניות. מערכות שונות יכולות להשתמש בתבניות לקסיקוגרפיות כדי לתת תפקידים סמנטיים לחלקי משפטים ולהוציא מהם תשובה אפשרית. *shallow semantic parsing* הוא מיתוג חלקים במשפט לתפקידים סמנטיים, דהיינו בעלי משמעות מסויימת באופן יחסי למילת יעד (ידוע גם כ- *semantic role labeling*). שיטה זו יכולה להתבסס למשל על *VerbNet* כדי לפענח תפקיד סמנטי של פעלים במשפט.
  - ***Semantic relation matching***: מעבר להסתכלות על מילים בודדות או אוספים של מילים, חיזוק משמעותי להוצאת תשובה יכול להתקבל ע"י התאמת קשרים סמנטית בין מרכיבים שונים.
  - ***Logical inference***: מיפוי משפטים לצורה פורמלית. *Deep semantic analysis* הוא הצגת משפטים בצורה לוגית פורמלית שניתן לבטא היקשים לוגיים פורמליים (למשל: לוגיקה מסדר ראשון). ע"י פורמליזציה של השאלה ושל מקורות התשובה ניתן להגיע לתשובה ע"י הסקת מסקנות. כל נושא הסקת המסקנות בכלל, כלומר היכולת לגזור תשובה שאינה מצויה באופן ישיר בטקסט ע"י היקש מעובדות שכן קיימות בטקסט הוא נושא מורכב וחשוב בניתוח סמנטי לצורך מציאת תשובה ובכלל ב*AI*.

השימוש ב-*Logical inference* יכול להיות מבוסס סינטקטית לחלוטין, אך כמובן בכדי לחזק את יכולות מערכת כזו ונכונותה יש צורך בכלים נוספים לניתוח. דוגמאות:

- מערכות המדרגות קשרי הסקה בין זוגות מילים המבוססים על *hypernym* ו-*synsets* מ-*WordNet*. למשל, ניתן להסיק כי *T* גורר את *H* אם כל ה-*hypothesis-concepts* של *T* הם *hypernyms* או *synonyms* של כל אלו של *H*.
- *Word sense disambiguation* כדי לא לאבד משמעות סמנטית בעת התהליך שיגרמו להסקת מסקנות שגויה.
- שימוש ב-*World Knowledge* כדי לבנות *"world knowledge axioms"* לחיזוק תהליך ההסקה. מבוסס למשל על יחסים סמנטיים מתוך *WordNet*.
- *Coreference resolution*: כמובן שהכרחי בשלבים ראשונים כדי לאגור את כל המידע על כל הפרטים המרכיבים את הטקסט.
- ובכלל עוד כלים חשובים: *named entity resolution, POS tagging* וכו'.

- ***Definitions***: כאשר מדובר בהגדרות מערכות שונות משתמשות במקורות מידע כגון *WordNet* המספקות שירותי *thesaurus* ו-*dictionary*.
- ***Clarfications***: ישנן מערכות המשלבות מנגנונים להתמודדות עם שאלות עמומות ע"י שאילת המשתמש שאלות תוך כדי מיקוד אזור החיפוש. בהתאם לתשובות המשתמש נמשך החיפוש לאזור המתאים שיכיל את התשובה.

לסיכום, כדי להשיג תוצאות טובות, גם ובעיקר בשימוש ב-*logical inference*, יש צורך בשילוב כלי*NLP* שונים יחד.

שקופית 6: *A brief history of QA*

תחום ה-*QA* אינו תחום מחקר חדש; מערכות ה- *QA* הראשונות שהופיעו כבר ב-1965 היו בעיקרן עבור *factoid questions*, דהיינו שאלות שהתשובה אליהן יכולה להסתכם לכדי עובדה, כגון שם, תאריך או מספר. רוברט פ. סימונס מאוניברסיטת טקסס, שהיה בעל תואר *PhD* בפסיכולוגיה, חקר את רשתות סמנטיות ובנה בעזרת תלמידיו מספר מערכות *QA* לייצור והבנת שפה טבעית. חזונו היה שהאדם יוכל "לשוחח עם ספר", כלומר שמחשב יוכל לקרוא ספר ולאחר מכן לנהל שיחה עליו עם אדם.

תחום ה-*QA* משולב בתחומי מחקר רבים ב-*NLP*, למשל:

- מסדי נתונים מבוססי *NL*
- מערכות דיאלוג: מערכות מחשב בעלי יכולת ניהול שיחה עם אדם במבנה מסויים. למשל, הדוגמא שהביא פרופ' דרשוביץ על אותו אדם ששוחח עם המחשב וחשב שהוא משוחח עם אדם אחר.
- *Reading comprehension systems*: מערכות מחשב המסוגלות לעבד טקסט בשפה טבעית ולענות על שאלות לגביו.
- *Open domain QA*: דובר

כמובן ש-*QA* היא תחום חשוב המשתלב בתחומים נוספים פרט לאלה, אך אלו מבוססים לחלוטין על *QA* לעומת תחומים שרק משלבים טכנולוגיית *QA* בין יתר הטכנולוגיות.

שקופית 7: *NLDBS*

מערכות מסדי נתונים מבוססי שפה טבעית הן מערכות שמקבלות שאלות ומתרגמות אותן לשאילתת מסדי-נתונים, למשל הדוגמא שבמצגת: תן את רשימת כל השחקנים שצברו מעל600 נקודות בכל המשחקים בשנת2009.

מערכות אלו הן מהמערכות הראשונות שנעשה בהן שימוש בטכנולוגיית*QA*, שכן אלו מערכות שכל ייעודן הוא לתת נתונים כמענה לשאלות, או שאילתות ליתר דיוק, ולכן טבעי היה ליישם תחילה טכנולוגיית*QA* במערכות אלו. שתי מערכות לדוגמא שפותחו בשנות ה-60:

BASEALL: מערכת שענתה על שאלות על משחקי בייסבול מהליגה האמריקאית בתקופה של עונה.

LUNAR: מערכת שפותחה כדי לתת מענה נוח לגיאולוגים החוקרים את ההרכב הכימי והגיאולוגי של הירח, לאחר משימת אפולו שאספה דגימות מהירח. בכנס מחקר הירח ב-1971 הוצגה המערכת ומתוך 111 שאלות לא-קומפרטיביות בנושא ידעה לענות נכון על78% מהשאלות.

המערכות הנ"ל הן דוגמאות שנחשבו מתוחכמות אפילו בסטנדרטים של היום.

שקופית 8: *Dialog Systems*

מערכות המאפשרות קיום שיחה עם מחשב, בין היתר מענה על שאלות.

מערכת ה-*SHRDLU* שפותחה ב- *MIT* בסוף שנות ה- 60 היתה מערכת ב- *closed domain* שדימתה עולם וירטואלי עם צורות גיאומטריות פשוטות. המערכת אפשרה למשתמשים לשאול שאלות לגבי המצב של העולם. למשל בדוגמת הטקסט במצגת. מערכת זו היתה פשוטה במובן שהיתה סגורה לסט כללים פיסקליים שקל ליישמו בתוכנת מחשב, כמו גם לעולם מצומצם של פרטים פשוטים.

ELIZA היא מערכת שפרופ' דרשוביץ הזכיר בשיעור הראשון, שבאה לדמות שיחה עם פסיכולוג וירטואלי. המערכת יכלה לדמות שיחה על כל נושא שהוא, בכך שהשתמשה בסט חוקים פשוטים לפיהם משכה מילים חשובות ממשפטי המשתמש ובנתה מענה. *ELIZA* נראית מערכת פחות מתוחכמת מ-*SHRLDU*, אבל זה רק כיוון ש-*SHRLDU* עובדת היטב תחת עולם מושגים מצומצם.

ELIZA פתח תחום של *CHATTERBOTS*, שהם תוכניות מחשב שאמורות להיות בעלי יכולות ניהול שיחה עם בני אדם. מדי שנה נערכת תחרות *AI* על פרס *Loebner* בה שופטים משוחחים דרך מחשב עם תוכנת*CHATTERBOT* ועם אדם, ועליהם לקבוע מי זה מי.

שקופית 9: *Reading comprehension systems*

מערכות *Reading Comprehnsion* הן מערכות המסוגלות לעבד טקסט ולהבין אותו, תוך יכולת לענות לאחר מכן על שאלות לגביו.

במערכות הבאות לבחון הבנת הנקרא משתמשים כדי לבחון יכולות קריאה והבנה בסיסיות בבתי ספר, כמו גם כדי לבחון את מידת ההבנה של מערכות ממוחשבות המיועדות לטפל בלימוד והבנת מידע.

QUALM שפותחה ע"י *Lenhert* בשנת 1978 היתה מערכת *QA* בעלת יכולת לבנות בסיס ידע תוך כדי הבנת סיפורים, והיכולת לענות על שאלות לאחר מכן.

ה-*DEEP READ* היא מערכת חדשה יותר שמסוגלת ע"י שימוש ב- *pattern matching* יחד עם כלים שונים לעיבוד שפה כגון *stemming, name identification, part-of-speech tagging, semantic class identification, pronoun resolution* למשוך ב-30-40% מהפעמים משפט מהטקסט בו מופיעה התשובה הנכונה. המערכת מתבססת על סיפורים ברמת כיתה ג'-ו' (בארה"ב).

לעומת מערכות *open-domain*, שם התשובה יכולה להופיע במסמכים רבים ובמקומות רבים, במערכות כגון הנ"ל סביר שהתשובה תופיע במקום אחד בלבד, ולכן משימת מתן התשובה במערכות כאלה קשה יותר.

שקופית 10: *Open Domain QA*

במערכות *QA* ב-*open domain* אין הגבלות על המשתמש להשאר תחת עולם מצומצם, אלא באפשרות המשתמש לשאול שאלה בכל תחום שהוא רוצה. כיוון שאין הגבלה על תחום השאלה במערכות *Open domain*, מערכות אלו משתמשות באוסף נרחב של מסמכים מתוכו נעשה הניסיון להוציא את התשובה הנדרשת. כיוון שתחום חיפוש התשובה נרחב, סביר כי לשאלה כלשהו תמצא יותר מתשובה אחת – עובדה זה יכולה להקל על מערכות *QA* בהפחתת הצורך בכלי עיבוד *NL* מסובכים.

לפני שהרשת נכנסה לשימוש רחב, השימוש היחיד ב-*QA* היה במערכות *closed domain*. השימוש ברשת בעל יתרונות חזקים:

- ה-*WWW* מכילה כמות אדירה של מסמכים, וכל אדם יכול לפרסם בה מה שהוא רוצה, כך שהסיכוי למצוא תשובה לשאלה בתחום כלשהו גדול יותר מאשר כל מאגר נתונים אחר.
- מתעדכן תמיד: גם שאלות על נושאים חדשים ימצאו תשובה בסיכוי טוב לעומת מערכות *Closed domain* שם לא סביר כי תמצא תשובה.

אך עם זאת ישנן בעיות:

- כיוון שאין הגבלה לאיזה חומר עולה לרשת ואיזה לא, ניתן להתקל בנתונים כוזבים ובסופו של דבר תשובה שקרית, ללא דרך לאמת זאת. עם זאת, הימצאות תשובות רבות לשאלה בתחום פתוח מאפשרות בחירת תשובה שסטטיסטית סביר כי היא הנכונה ולפיכך המבוקשת.
- לא תמיד ניתן לדעת את תוקף המסמכים שנמצאים ברשת וזהו פתח לתשובות לא עדכניות. למשל, על השאלה "מי נשיא ארה"ב?" ניתן לקבל יותר מתשובה אחת (טוב, אולי לא השאלה הזו...)

שקופית 11: *TREC*

TREC, או *Text retrieval conferences* היא סדרת סדנאות שנערכת מידי שנה מאז 1992, ממומנת ע"י ה-*NIST* (*national institute of standards and technology*) ומשרד ההגנה האמריקאי, שמטרתה לעודד מחקר בתחום ה-*IR* המבוסס על אוספי טקסט גדולים. הכנסים מהווים מקום מפגש של התעשיה והאקדמיה בתחום ויוצרים פורום שיתוף מידע ורעיונות.

מדי שנה *NIST* מספקת למשתתפי *TREC* אוסף מסמכים ושאלות בתחום משתנה. המשתתפים מריצים את מערכות ה-*IR* שלהם על אותו אוסף מסמכים ושאלות, והתוצאות עוברות ל-*NIST* לשיפוט נכונות והצלחה. לבסוף נערכת סדנה עם כל המשתתפים כפורום שיתוף.

ב-*TREC* נעשה מיקוד בכל תחומי המחקר הרלוונטיים בתחום ה-*IR*, למשל הוצאת מידע רלוונטי למחקר הגנום הכולל רצפי גנום לצד מחקרים ותוצאות מעבדה רלוונטיים, פיתוח טכנולוגיות חיפוש עבור משפטנים למציאת מידע אפקטיבית במסמכים אלקטרוניים, ותחומים רבים אחרים שמשתנים מדי שנה ושנה. אחד התחומים הוא כמובן *QA*, הורץ פעם אחרונה ב- *TREC* 2007, כאשר המטרה היא מענה נכון ומדוייק על שאלות מבוססות עובדות, רשימות והגדרות.

מדי שנה עולה רמת הקושי של המסמכים והשאלות, כמו גם הדרישות מהמערכות המשתתפות, למשל מעבר מדרישת 5 תשובות אפשריות מדורגות לדרישת תשובה מדוייקת יחידה, שאלות ומסמכים המשקפים יותר ויותר טוב את ה"עולם האמיתי" (למשל, חיפוש בבלוגים בהם השפה פחות מדוייקת ומובנית), שאלות שהתשובות עליהן לא בהכרח מצויות במסמכים וכו'.

שקופית 12: *Question Answering based on Semantic Graphs*

המערכת שאציג כעת היא פרי פיתוח קבוצת מחקר במחלקת טכנולוגיות מידע במכון *Josef Stefan* בסלובניה, שהוצגה בכנס ה-*WWW* שנערך באפריל 2009 במדריד. המערכת היא מערכת *QA* מבוססת על גרפים סמנטיים, המספקת בנוסף על תשובות גם הסברים לתשובות מתוך ייצוג גרפי של המסמכים, רשימת העובדות המקושרות למסמכים המתוארת ע"י שלשות נושא-נשוא-מושא, וסיכומם.

שלישיות הנושא-נשוא-מושא מחולצות מתוך המשפטים המרכיבים את אוסף המסמכים וניתן לערוך בהן חיפוש דרך יישום *QA*. מערכת זו לא מוגבלת לתחום אחד, כלומר היא *open domain QA system*, אבל כן מוגבלת דקדוקית לסוגי שאלות מסויימים. עיבוד המסמך נעשה *offline* וכרגע המערכת יושבת על מסד הנתונים של *reuter-news*. התשובות נלקחות מרשימת עובדות ונתמכות ע"י משפטים והמסמכים מהם באו.

שקופיות 13-14: מוטיבציה

הדוגמא שעליה נראה את שאר ההצגה ולבסוף גם נערוך עליה הדגמה היא השאלה *where do tigers live*, כאשר המסמך שיבחר הוא מסמך המדבר על נמרים בסומטרה (אינדונזיה) ושילובם ברפואה הסינית.

המערכת מספקת פונקציונאליות חשובה:

1. סיפוק תשובה לשאלה
2. סיור נוח ואינפורמטיבי במסמך ממנו נלקחה התשובה: רשימת עובדות מתוך המסמך (שלישיות נושא-נשוא-מושא), הצגת המסמך כגרף סמנטי וסיכום המסמך.

ניתן לראות בדוגמא הזו את הייצוג הסמנטי של המסמך שמתאר קשרים היוצאים מ ונכנסים אל ה-*sumatran tiger*, הצגה לוגית ונוחה להוצאת מידע מהמסמך.

יתרון משמעותי של מערכת זו בניגוד למערכות עבר היא השילוב של מספר כלים להוצאת מידע מהמסמך, כלים שקיימים במערכות אחרות מהעבר אך כ-*standalones*. כאן ניתנות למשתמש תשובות לצד אמצעים נוספים להבנת התשובה וקבלת מידע רלוונטי.

שקופית 15: *System Overview*

בשרטוט זה ניתן לראות בגדול את תהליך העבודה עם המערכת:

- המשתמש שואל שאלה בשפה טבעית
- לאחר חיפוש, המערכת מחזירה תשובה אפשרית המקושרת למשפטים תומכים במסמך ולמסמך עצמו.
- בנוסף מתקבל ה-*Document Overview* כולל רשימת עובדות, גרף סמנטי וסיכום.

השאלות יכולות להיות בכל תחום, ולא מוגבלות ל-*domain* מסויים, כלומר זו מערכת *open-domain*, שכרגע מבוססת על מאגר המידע של *reuter-news*. המידע אינו נלקח מתוך אונטולוגיות קיימות, אלא טקסט שאינו מובנה. ההגבלה הקיימת היא סוגי שאלות מסויימים.

בעיבוד המסמך עצמו נשלפות עובדות, *named entities extraction*, טיפול ב-*coreferences*, *anaphora* והתבססות על *WordNet* לנירמול סמנטי. שיטות אלו משמשות לבניית הגרף הסמנטי בו הצמתים הם מרכיבי השלישיות (ניתן להסתכל על זה גם כצמתים=נושא, מושא וקשתות=נשוא).

יצירת הסיכום נעשית ע"י שימוש בעובדות שנמשכו מהמסמך, המסמך עצמו והגרף הסמנטי.

בהמשך יורחב בקצרה על כל אחד מרכיבי המערכת הנ"ל.

שקופית 16: *Triplets*

השלישיות הן אבני הבניין של המסמך, המייצגות את המידע המאוסן במשפטים המרכיבים את המסמך. השלישיות המוצאות מהמסמך נשמרות לצורכי שליפה מהירה. השלישיות מהוות את הבסיס לבניית הגרף הסמנטי, כפי שיוצג בהמשך.

שקופית 17: *Question Answering*

שלב השאלה-תשובה עצמו מורכב מכמה שלבים:

- תחילה נעשית הוצאת השלישיות מהמסמך, כפי שהוזכר.
- השלישיות נשמרות לצורכי חיפוש.
- שאלות המתקבלות עוברות ניתוח לבניית השאילתא המתאימה לחיפוש השלישיה שתבנה את התשובה.
- התשובה מוחזרת ויחד איתה המסמך ממנו הוצאה.
- ניתן לסרוק את ה-*document overview* לקבלת מידע נוסף

שקופית 18: המשך...

1. המערכת משתמשת במנוע חיפוש של *Text-Garden* המספק כלי חיפוש מתקדמים. המנוע תומך בחיפוש שלישיות כאשר אחד או יותר מגורמיה חסר, למשל: החזר את כל השלישיות בהן הנושא הוא *tiger* והנשוא הוא *live*. ניתן לבצע חיפוש עם אטריביוטים מיוחדים, כגון דרישת *negation* על הנשוא, למשל *tiger+not eat+ people*.
2. בשלב ניתוח השאלה נוצרת השאילתא עצמה שתרוץ במאגר השלישיות. השאילתא כאמור נבנית ע"י *subset* של שלישיה, ובנוסף נעשה שימוש ב-*WordNet* להוצאת *synonyms* להרחבת השאילתא, למשל בדוגמא תשאל השאלה הנוספת *tiger+inhabit+?*.
3. בניתוח השאלות נקבע סוג השאלה ובהתאם סוג השאילתא שתורץ במנוע החיפוש.
4. השאילתות מתחלקות לשלושה סוגים:
5. *LeafQuery*: שאילתא בסיסית המשמשת לחיפוש ישיר של שלישיית נושא-נשוא-מושא.
6. *UnionQuery*: למעשה היא איחוד של כמה שאילתות בסיס, מחזירה את איחוד של השלשיות לתת השאילתות שמכילה.
7. *FilterQuery*: צמצום שאילתא בסיסית ע"י העברתה בפילטר מסויים על גבי אחד או יותר מחלקי השלישיה, למשל: הנושא חייב להיות מספרי; הפועל חייב להיות בצורה *negation* וכדומה.

שקופית 19: *Question types*

סוגי השאלות בהם המערכת תומכת הן:

- *yes/no*
- שאלות שהתשובה להן היא רשימת אובייקטים
- שאלות סיבתיות
- שאלות כמותיות
- שאלות מיקום או זמן

ניתוח השאלה עצמו מתחיל ב-*parsing* של השאלה הבונה *treebank* לפי פורמט *upenn treebank* שפותח באונ' פנסילבניה. בשלב זה חלקי השאלה/עץ מקבלים תיוג בהתאם לתפקידם התחבירי, למשל שמות עצם, פעלים, *wh-adverbs* (*wh-questions*) ועוד. בשלב זה נעשית גם ההרחבה של השאילתא ע"י *synonyms* ו-*hyponyms* (שמות מוכללים) המסתמכים על מאגר הסטים ב-*WordNet*.

בניית התשובה היא בניית סט של שלישיות בהתאם לסוג השאלה:

- **שאלות *y/n***: יחולקו לשני סטים: סט אחד בו הפועל הוא ב- *polarity* זהה לזה שבשאלה (מבחינת *negation*), ולסט שני בו ה-*polarity* הפוך לזה שבשאלה. הסט הראשון יחשב תשובת *yes* והשני תשובת *no*, ואלו ייתמכו ע"י המשפטים מהם הוצאו השלישיות.
- **שאלות כמותיות/רשימות/מיקום**: התשובה תורכב מאובייקטים רבים, אלמנטים רלוונטים שיחולצו מתוך השלישיות המוחזרות. אלמנטים אלו יקובצו יסודרו לפי סדר יורד של תדירות ההופעה שלהם, וכל מקבץ אלמנטים הוא פריט תשובה שיוחזר בסוף.
- **שאלות זמן/סיבה**: במקרים אלו לא תיבנה תשובה מיוחדת,אלא המשפט המכיל את השלישיות המוחזרות מהשאילתא פשוט יוחזר כפי שהוא כתשובה.

שקופיות 20-22: *screen shots*

סקרין שוטים:

- שאלה – מקבלים תשובה בצורת שלישיות, ורשימת המסמכים
- בבחירת מסמך ניתן לראות את המסמך ולחקור אותו:
- גרף סמנטי, רשימת עובדות וסיכום

בסוף אם יהיה זמן נראה דמו

שקופית 23: *Document Overview*

- תחילה מדגישים את העובדות במסמך המיוצגים ע"י שלשות.
- משתמשים בשלשות לבניית הגרף הסמנטי שהוא הייצוג הסמנטי של המסמך.
- בונים את הסיכום של המסמך.

שקופיות 24-27: *Semantic Graph*

שלבי בניית הגרף הסמנטי המייצג של המסמך הם אלו:

- תחילה המסמך מחולק למשפטים

- לאחר מכן נעשה *named entity extraction*, ו-*coreference resolution* כדי לאחד אינסטנציות המתייחסות לאותו שם/אובייקט יחד לקראת הוצאת השלישיות. בדוגמא ניתן לראות זיהוי של*ASIA* כמיקום ו-*WWF* והופעות נוספות המתייחסות אליו כארגון.

<u>פירוט על *named entity extraction*</u>:

- השמות המוצאים הם של אנשים, מקומות, ארגונים, כל אלו פריטי מידע סמנטי חשוב.
- המערכת עושה שימוש בכלים קיימים לזיהוי יישויות, כאשר עבור אנשים נשמר המגדר, עבור מקומות – עיר/ארץ.
- באמצעות כלי ניתוח טקסט ושיטות השוואה נעשה *co.ref. res.*, למשל שמות המוכלים אחד בשני (*Joe, Joe Black*), ראשי תיבות וכד'. לרוב מסירים *stop words*.
- השלב הבא הוא הוצאת השלישיות מתוך המשפטים.

<u>פירוט על הוצאת שלישיות</u>:

- הנחת עבודה: השלישיות מהוות תמצית מספקת להעברת המסר הגלום במשפט.
- פשטות השלישיות מקלה על הניתוח לעומת משפטים שלמים מורכבים.
- השלישיות מוצאות מתוך כל משפט באופן בלתי תלוי בטקסט מחוצה לו על בסיס ניתוח סינטקטי בלבד.
- מבנה ה-*parse tree* של המשפטים מהווה בסיס להחלטה אילו שלישיות להוציא.
- בנוסף לשלישיות מוציאים גם תכונות שונות המקושרות להן, למשל מילים המקושרות למושא ב*parse tree*.
- לאחר מכן נעשה *enhancement* לחלוקה לשלישיות ע"י *pronominal anaphoras resolution* וקישור כל שלישיה ל-*syset* שלה ב-*WordNet*.

<u>פירוט על ה-*enhancement*</u>:

- *Anaphora resolution* נעשית על בסיס ה-*coreference resolution*, ונעשה חיפוש להחלפת *pronouns* בשם ישיר, כאשר המועמדים להחלפה מדורגים על פי מס' חזרות, מרחק מכינוי הגוף שמתייחס אליהם, תבניות צירופי לשון וכד'.
- בשלב השני השלישיות מקושרות ל-*synset* המתאים להם מ-*WordNet*. שלב זה מגביר את היכולת למזג אובייקטים מאותו *synset* ובעלי משמעות דומה ובכך מעדן את הגרף.
- השלב האחרון הוא מיזוג של כל השלישיות לכדי גרף מכוון אחד, כאשר הכיווניות שלו היא מהנושא למושא כאשר הנשוא הוא חוליה מקשרת בין השניים.

שקופיות 31-28: *Document Summary*:

1. הסיכום נבנה על בסיס המסמך המקורי והייצוג הסמנטי שלו, ומורכב ממשפטים מהטקסט המקורי תוך שמירה על סדר ההופעה.
2. המערכת מוציאה מהמסמך מאפיינים על כל המשפטים המרכיבים אותו הכוללים תכונות לשוניות, תכונות של המסמך או תכונות של הגרף הקשורות לשלישיות.
3. ע"י אימון *linear SVM* (*support vector machine*) מתבצעת קלאסיפיקציה של השלישיות שהמשפטים מהם נגזרות יהוו חלק מהסיכום.
4. סיכומים אפשריים נוצרים תוך שמירה על סדר המשפטים המרכיבים במסמך, מדורגים לפי ה-*score* שלהם ב-*SVM* ומסודרים בסדר יורד לפי נתון זה.

לבסוף נעשית הערכה של טיב הסיכום המבוססת על*datasets* מה-*DUC* (*document understanding conferences*) – בדיקת וידוא איכות הסיכום.

שקופיות 33-32: המשך...

ובדוגמא שלנו...

שקופית 34: *Conclusions*

אז מה היה לנו? מערכת *QA* מתקדמת שמספקת תשובות מדוייקות ולצדן מסמכים תומכים והאפשרות לסייר בעובדות המוצגות במסמך, ייצוג סמנטי של המסמך וסיכום.

שקופית 35: *References*