CS585 Assignment 3: Report

Ganesh Mahesh

Department of Computer Science

Illinois Institute of Technology

May 5,2017

Abstract

Implementation of lexical chains in Java with the help of Wordnet and JWNL (Java Wordnet Library) and text summarization using word frequency count in chain.

Problem statement

- Relation among different parts of a text plays a vital role in understanding the information present in the text
- Nouns and Pronouns significantly help in understanding the topic of a text
- Extracting those words which are similar in sense is vital for this purpose
- Summary of the text depends on the word usage and its relations in the text

Proposed solution

Over view:

- Scan the text word by word
- Identify the relation among already scanned words and the new word being scanned
- Represent the relation of words by means of lexical chains
- Lexical chains represent the discourse structures present in the text under consideration
- From Lexical chains summary is constructed
- Wordnet is used to identify different senses of the word and to identify the relation among words
- JWNL acts as an interface to Wordnet.

Current Implementation

- Input
    - Text file named **sentences** containing the text needed to be processed
- Scan text file word by word
- Obtain the Synsets (different senses) associated with the word under consideration

NOTE: scanning through the input file generates a structure of data having word, sentence in which the word occurred and its line number. These information is used while sorting and creating the summary

- Form lexical chain from the relation among words
    - First word, scanned will form a chain by itself
    - Further words being scanned will have its synsets compared with the words already in chains
    - Relation comparison done on the basis of (Antonym, Hypernym, Hyponym and Synonym)

  - If a word relates to an existing chain element, addition of the word to the respective chain is carried out (if word already present frequency counter is increased for that chain element)
- From lexical chain construct summary
  - Each element in chain has frequency associated with it
  - Among the created chains, highest frequency element from each chain is extracted
  - This list of highest frequency elements are sorted in descending order such that highest is at the top of the list
  - Top 5 elements are chosen for creating summary
  - These 5 elements are sorted in ascending order based on the line number associated with the word
  - Among these 5 elements from the first element summary is constructed such that no sentence is repeated
- JWNL acting as an interface to Wordnet helped in traversing wordnet database to fetch necessary relation of words.
  - Main methods used are `findRelationships taking in parameter of Relation type (HYPERNYM, HYPONYM and ANTONYM)`
  - Synset i.e different senses of the particular word under consideration are extracted using `dictionary.lookupIndexWord`

Future Enhancements

- Word sense relation mapping can be better handled by using more lexical relationships
- Restricting the addition of word to chain by making sure the sense of the new word is consistent with the chains lexical meaning
- Enabling Phrasal nouns addition to the chain
- Summary formation may consider location, cues and word frequency together for its betterment

Manual

1. Input through text file

sentences is the file in which the text needed to be analyzed should be placed

path of the sentence file: is hard coded and can be changed by editing the file LexicalChainFinder.java line 24

```
if (fileHandler.initialize("sentences") == false) {
```

in place of `sentences` need to mention the full path of the file which you want to use

sentences file is present in the project directory.

2. Output seen in console

Output is seen in the console once the project is built and run

Output format is of the type

Chain “number”: “word” “frequency of repetition”

Summary sentence

“line number”

Example Results for different text input

1.

Input text present in “sentences” file:

In linguistics, a hyponym (from Greek hupó, "under" and ónoma, "name") is a word or phrase whose semantic field is included within that of another word, its hyperonym or hypernym (from Greek hupér, "over" and ónoma, "name"). In simpler terms, a hyponym is in a type-of relationship with its hypernym. For example, pigeon, crow, eagle and seagull are all hyponyms of bird (their hypernym); which, in turn, is a hyponym of animal.

Output:

```
chain 1: hyponym(3), word(1), hypernym(1),
chain 2: Greek(2),
chain 3: phrase(1),
chain 4: field(1),
chain 5: relationship(1),
chain 6: eagle(1),
chain 7: seagull(1),
chain 8: are(1),
chain 9: bird(1),

In linguistics, a hyponym (from Greek hupó, "under" and ónoma, "name") is a word or
phrase whose semantic field is included within that of another word, its hyperonym or
hypernym (from Greek hupér, "over" and ónoma, "name").
1
```

2.

Input text present in “sentences” file:

A single-engine airplane crashed Tuesday into a ditch beside a dirt road on the outskirts of Albuquerque, killing all five people aboard, authorities said.

Four adults and one child died in the crash, which witnesses said occurred about 5 p.m., when it was raining, Albuquerque police Sgt. R.C. Porter said.

The airplane was attempting to land at nearby Coronado Airport, Porter said.

It aborted its first attempt and was coming in for a second try when it crashed, he said…

Output:

```
chain 1: airplane(2),
chain 2: Tuesday(1),
chain 3: ditch(1),
chain 4: dirt(1), land(1),
chain 5: road(1),
chain 6: outskirts(1),
chain 7: Albuquerque(2),
chain 8: killing(1),
chain 9: five(1),
chain 10: people(1),
chain 11: authorities(1),
chain 12: Four(1),
chain 13: one(1),
chain 14: child(1),
chain 15: crash(1),
chain 16: police(1),
chain 17: Porter(2),
chain 18: Airport(1),
chain 19: first(1),
chain 20: attempt(1), try(1),
chain 21: coming(1),
chain 22: second(1),

A single-engine airplane crashed Tuesday into a ditch beside a dirt road on the
outskirts of Albuquerque, killing all five people aboard, authorities said.
1
      Four adults and one child died in the crash, which witnesses said occurred
about 5 p.m., when it was raining, Albuquerque police Sgt. R.C. Porter said.
2
```

3.

Input Synthetic example just to show the relation categories being used and successfully being chained together:

white albumen case example
chordate animal case is a beast
light dark case example
sparkle twinkle night wickedness
sound of music is a movie played in cinema known for its cinematography
fish Pisces is a particular swimming animal with which fishing is carried out

```
chain 1: white(1), albumen(1),
chain 2: case(3), example(2),
chain 3: chordate(1), animal(2), beast(1),
chain 4: light(1), dark(1), sparkle(1), twinkle(1), night(1), wickedness(1),
chain 5: sound(1), music(1),
chain 6: movie(1), cinema(1),
chain 7: cinematography(1),
chain 8: fish(1), pisces(1),
chain 9: swimming(1),
chain 10: fishing(1),
chain 11: out(1),

white albumen case example
1
chordate animal case is a beast
2
light dark case example
3
sound of music is a movie played in cinema known for its cinematography
5
```

4. input

Having lost the first two games of their Eastern Conference semifinal series at home, Toronto made a lineup change as the series shifted to Cleveland for Game 3.

Fred VanVleet, Toronto's superb sixth man, entered the starting lineup, with Raptors Coach Dwane Casey choosing to bring Serge Ibaka off the bench instead.

The small lineup didn't exactly work as Casey intended, with Cleveland jumping out to a 16-4 lead.

But the Raptors managed to cut the Cavaliers' lead to 24-19 after one thanks to a 15-8 run to end the quarter, largely executed with Ibaka playing center.

After failing to get a block in either of the first two games of the series, Ibaka had two in the first quarter alone to help spark Toronto's run to get back into the game.

In a wild, wacky and thrilling game, the Celtics emerged with a heart-stopping 101-98 overtime victory over the Sixers in Game 3 of their Eastern Conference semifinal.

The win gives the Celtics a 3-0 lead in this best-of-seven series, a lead no team has ever given up in an NBA playoff series.

Boston looked like it won in regulation when Philadelphia committed a turnover that led to a Jaylen Brown dunk with 1.8 seconds left to give the Celtics a two-point lead.

But Marco Belinelli hit a jumper in the corner right in front of the Sixers' bench to tie the game and send it to overtime.

Philadelphia then looked like it was in control when it led by one with the ball with 42.5 seconds left.

But Joel Embiid missed a shot in the lane, and when Ben Simmons got the offensive rebound, he chose to go up for a putback instead of pulling the ball back out, and missed as the shot clock was turned off.

Boston then got the rebound and, after bringing the ball up and calling timeout, Marcus Morris threw a perfect inbounds pass to Al Horford, who had expertly sealed his defender, Robert Covington, at the rim, allowing Horford to catch the ball and lay it in to give the Celtics a 99-98 lead.

Output:

```
chain 1: lost(1),
chain 2: first(3), starting(1), end(1),
chain 3: two(3),
chain 4: Conference(2),
chain 5: semifinal(2),
chain 6: series(5),
chain 7: home(1),
chain 8: Toronto(1),
chain 9: lineup(3),
chain 10: change(1),
chain 11: Cleveland(2),
chain 12: Game(2), game(3), catch(1),
chain 13: sixth(1),
chain 14: man(1),
chain 15: Coach(1),
chain 16: Serge(1),
chain 17: bench(2),
chain 18: small(1),
chain 19: work(1),
chain 20: jumping(1),
chain 21: out(2),
chain 22: lead(6), cut(1), jumper(1), shot(2), ball(3), pass(1),
chain 23: one(2),
chain 24: thanks(1), help(1),
chain 25: run(2), spark(1),
chain 26: quarter(2), back(2), front(1),
chain 27: playing(1),
chain 28: center(1),
chain 29: failing(1),
chain 30: get(2),
chain 31: block(1),
chain 32: wild(1),
chain 33: overtime(2),
chain 34: victory(1), win(1),
chain 35: over(1),
chain 36: team(1),
chain 37: given(1),
chain 38: playoff(1),
chain 39: Boston(2),
chain 40: like(2),
chain 41: won(1),
chain 42: regulation(1), control(1),
chain 43: Philadelphia(2),
chain 44: turnover(1),
chain 45: led(2),
chain 46: Brown(1),
chain 47: dunk(1),
chain 48: left(2), right(1),
chain 49: give(2),
chain 50: hit(1),
chain 51: corner(1),
chain 52: tie(1),
chain 53: then(2),
chain 54: ball(1),
chain 55: Joel(1),
chain 56: lane(1),
chain 57: Ben(1),
chain 58: offensive(1),
chain 59: rebound(2),
chain 60: pulling(1),
chain 61: clock(1),
chain 62: bringing(1),
chain 63: calling(1),
chain 64: Morris(1),
chain 65: perfect(1),
chain 66: who(1),
chain 67: defender(1),
chain 68: Robert(1),
chain 69: rim(1),
chain 70: lay(1),
```

Having lost the first two games of their Eastern Conference semifinal series at home, Toronto made a lineup change as the series shifted to Cleveland for Game 3.

```
1
The small lineup didn’t exactly work as Casey intended, with Cleveland jumping out to
a 16-4 lead.
3
After failing to get a block in either of the first two games of the series, Ibaka
had two in the first quarter alone to help spark Toronto’s run to get back into the
game.
5
```

Results and discussion

- Input set 3 is synthesized to show all three relation criteria’s (synonym, antonym, hypernym, hyponym) being used to chain words
    - Chain 1 and 2 are synonym based
    - Chain 3 is based on hyponym/hypernym (chordate, animal) with depth of 1 and synonym (animal, beast)
    - Chain 4 is based on antonym (light, dark) and other synonym corresponding to these words
    - Fish though being animal is not categorized as such because the depth of its relation is more than 1
- Input set 2 similarly chins are formed one noticeable fact is words which appear more than once are grouped and are shown by the frequency count within parenthesis ( )
- Input set 1 also highlights the fact that seagull and eagle though being bird are not categorized under it because of the depth different seen more than 1. Improvements can be made by addition of other criteria’s and increasing the depth of relation traversal.
- Summary generated in each case can be observed to be coherent with the word with largest frequency