

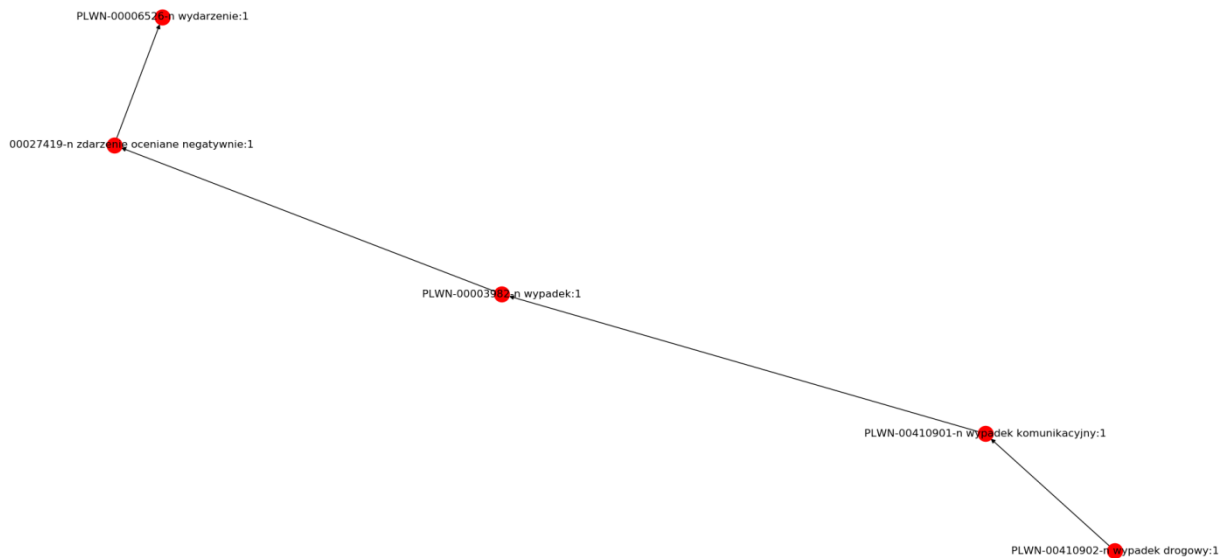
1. Znajdź wszystkie znaczenia **rzeczownika** *szkoda* oraz wymień ich synonimy (jeśli posiadają).

PLWN-00003675-n {strata:1, utrata:1, szkoda:2, uszczerbek:1} ()

PLWN-00006533-n {szkoda:1} (brak danych)

2. Znajdź domknięcie przechodnie relacji **hiperonimi** dla pierwszego znaczenia wyrażenia *wypadek drogowy* i przedstaw je w postaci grafu skierowanego.

```
senses = wn.lookupLiteral(literal, part_of_speech)
result = wn.lookupSense(literal, sense_number, part_of_speech)
sense_index = get_sense_index(senses, result)
if not senses:
    print("Literal not found\n")
else:
    #oss = wn.traceRelationOS(senses[sense_index].wnid, part_of_speech,
    relation)
    #result.extend(oss)
    root = Node(senses[sense_index].wnid)
    wn.traceRelationMine(root.id, part_of_speech, relation, root)
    for child in root.children:
        print(child.synonyms)
return root
```



3. Znajdź bezpośrednie **hiponimy** rzeczownika *wypadek₁*.

```
root = get_relation("wypadek", "n", "hyponym", 1)
print("direct hyponym")
for child in root.children:
    print(str(child.id) + " " + "{0}".format(child.synonyms))
```

PLWN-00001284-n ['zderzenie:2', 'kraksa:1']

PLWN-00006486-n ['kolizja:2']

PLWN-00016131-n ['karambol:1']

PLWN-00034688-n ['zawał:2']

PLWN-00034689-n ['tąpienie:1']

PLWN-00241026-n ['kapotaż:1']

PLWN-00258639-n ['wykolejenie:2']

PLWN-00389170-n ['zakrztuszenie:1', 'zachłyśnięcie:1', 'aspiracja:3']

PLWN-00410901-n ['wypadek komunikacyjny:1']

PLWN-00411618-n ['katastrofa budowlana:1']

PLWN-00436137-n ['wypadek jądrowy:1']

4. Znajdź **hiponimy drugiego rzędu** dla rzeczownika *wypadek*₁.

```
print("2nd level hyponym")
get_hyponyms(root, 1)
def get_hyponyms(parent, level):
    if level == 0:
        for child in parent.children:
            print(str(child.id) + " " + "{0}".format(child.synonyms))
    for child in parent.children:
        get_hyponyms(child, level-1)
```

PLWN-00235346-n ['czołówka:9', 'zderzenie czołowe:1']

PLWN-00471555-n ['stłuczka:1']

PLWN-00441365-n ['kolizja drogowa:1']

PLWN-00037295-n ['obwał:1']

PLWN-00410902-n ['wypadek drogowy:1']

5. Przedstaw w postaci grafu skierowanego (z etykietami dla krawędzi) relacje semantyczne pomiędzy następującymi grupami leksemów:

```
result = []
for i in range(0, len(group)-1):
    isSynonym = False
    isHyponym = False
    for j in range(i + 1, len(group)):
        print(group[i][0] + " and " + group[j][0])
```

```

synset1 = wn.lookupSense(group[i][0], group[i][1], "n")
synset2 = wn.lookupSense(group[j][0], group[j][1], "n")
# bliskoznacznosc
try:
    foundtarg = wn.isIDConnectedWith(synset1.wnid, "n",
"bliskoznacznosc", synset2.wnid)
    if foundtarg:
        result.append((group[i][0], group[j][0],
"bliskoznacznosc"))
        continue
except RecursionError as re:
    print("bliskoznacznosc - recursion error")

# hiponimia
try:
    foundtarg = wn.isIDConnectedWith(synset1.wnid, "n",
"hiponimia", synset2.wnid)
    if foundtarg:
        result.append((group[i][0], group[j][0], "hiponimia"))
        continue
except RecursionError as re:
    print("hiponimia - recursion error")

# hiperonimia
try:
    foundtarg = wn.isIDConnectedWith(synset1.wnid, "n", "hypernym",
synset2.wnid)
    if foundtarg:
        result.append((group[i][0], group[j][0], "hiperonimia"))
        continue
except RecursionError as re:
    print("hiperonimia - recursion error")

```

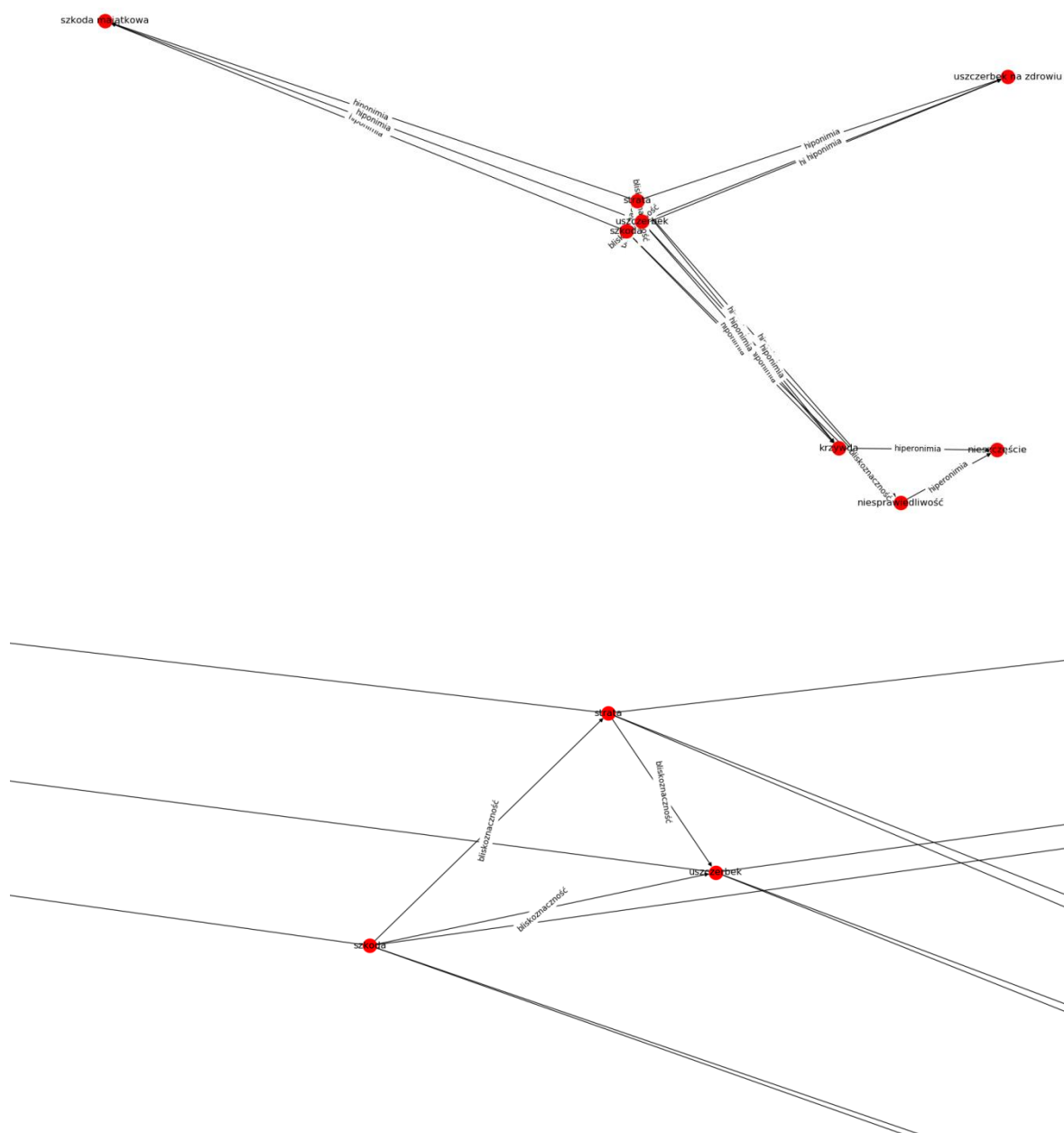
I

('szkoda', 'strata', 'bliskoznacznosc')
('szkoda', 'uszczerbek', 'bliskoznacznosc')
('szkoda', 'szkoda majatkowa', 'hiponimia')
('szkoda', 'uszczerbek na zdrowiu', 'hiponimia')
('szkoda', 'krzywda', 'hiponimia')
('szkoda', 'niesprawiedliwosc', 'hiponimia')
('strata', 'uszczerbek', 'bliskoznacznosc')
('strata', 'szkoda majatkowa', 'hiponimia')
('strata', 'uszczerbek na zdrowiu', 'hiponimia')
('strata', 'krzywda', 'hiponimia')
('strata', 'niesprawiedliwosc', 'hiponimia')
('uszczerbek', 'szkoda majatkowa', 'hiponimia')
('uszczerbek', 'uszczerbek na zdrowiu', 'hiponimia')
('uszczerbek', 'krzywda', 'hiponimia')
('uszczerbek', 'niesprawiedliwosc', 'hiponimia')

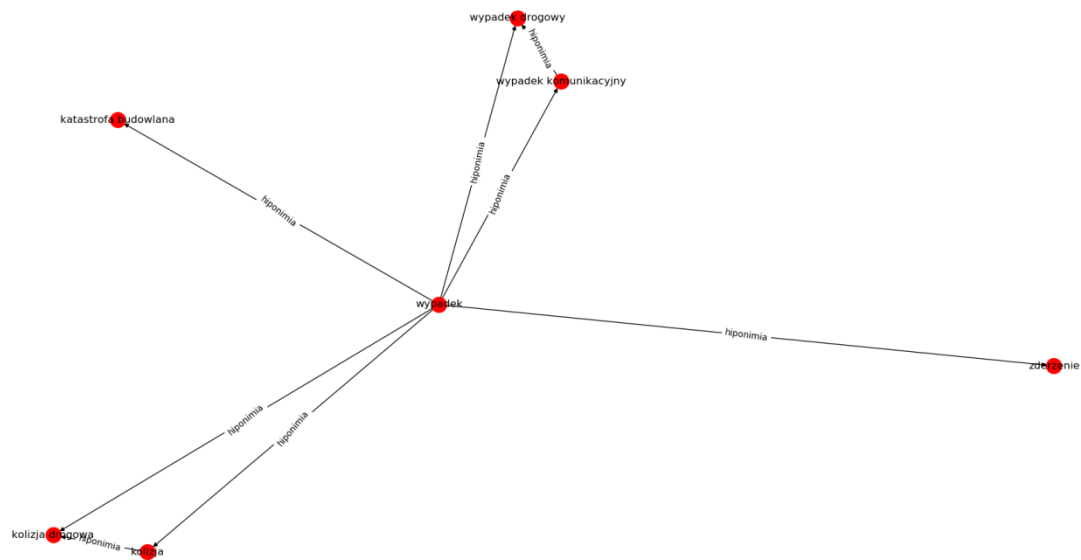
('krzywda', 'niesprawiedliwość', 'bliskoznaczność')

('krzywda', 'nieszczęście', 'hiperonimia')

('niesprawiedliwość', 'nieszczęście', 'hiperonimia')



II



6. Znajdź wartość miary pokrewieństwa semantycznego Leacock-Chodorowa ftp://www-vhost.cs.toronto.edu/public_html/public_html/pub/gh/Budanitsky+Hirst-2001.pdf pomiędzy następującymi parami leksemów: