# ДОДАТОК А

## Текст програми «KnowGraph»

```
import re
import pandas as pd
import spacy
from tqdm import tqdm
nlp = spacy.load("en_core_web_sm")
from spacy.matcher import Matcher
import networkx as nx
import matplotlib.pyplot as plt
import nltk
nltk.download('punkt')
from nltk.tokenize import word_tokenize
nltk.download('stopwords')
from nltk.corpus import stopwords
import re
import string
from google.colab import drive

from google.colab import files
up = files.upload()

text_corpra = pd.read_csv("wiki_sentences_v2.csv")
text_corpra.shape

def extract_entity(sent):
  subject = ""
  object = ""

  prev_chunk_text = ""
  prev_chunk_dep = ""
```

```
    pref = ""
    mod = ""

    for chunk in nlp(sent):
      if chunk.dep_ != "punct":
        if chunk.dep_ == "compound":
          pref = chunk.text
          if prev_chunk_dep == "compound":
            pref = prev_chunk_text + " "+ pref
        if chunk.dep_.endswith("mod")==True:
          mod = chunk.text
          if prev_chunk_dep == "compound":
            mod = prev_chunk_text + " " + mod

        if chunk.dep_.find("subj") == True:
          subject = mod + " " + pref + " " + chunk.text
          pref = ""
          mod = ""
        if chunk.dep_.find("obj") == True:
          object = mod + " " + pref + " " + chunk.text
          pref = ""
          mod = ""

        prev_chunk_text = chunk.text
        prev_chunk_dep = chunk.dep_

    return [subject.strip(), object.strip()]

def extract_relation(sent):

    doc = nlp(sent)

    match = Matcher(nlp.vocab)
```

```
  pattern = [{'DEP':'ROOT'},
             {'DEP':'prep','OP':"?"},
             {'DEP':'agent','OP':"?"},
             {'POS':'ADJ','OP':"?"}]

  match.add("matching_1", None, pattern)

  matches = match(doc)
  k = len(matches) - 1

  span = doc[matches[k][1]:matches[k][2]]

  return (span.text)

def stopWordCheck(word):
    return word not in stopwords.words('english')

def text_preprocessing(sent):
  sent = re.sub("[\(\[].*?[\)\]]", "", sent)
  chunks = []
  temp = ""
  words = word_tokenize(sent)
  punctuations = '"#$%&\'()*+,-/:;<=>@\\^_`{|}~'
  words = map(lambda x: x.translate(str.maketrans('','',punctuation
)), words)
  words = map(str.lower,words)
  words = filter(lambda x: stopWordCheck(x),words)
  chunks = chunks + list(words)
  temp = ' '.join(word for word in chunks)
  return temp

preprocessed_sentences = [text_preprocessing(i) for i in tqdm(text_c
orpra['sentence'])]
tqdm._instances.clear()
```

```
entity_pairs = []
relations = []
for i in tqdm(preprocessed_sentences):
  entity_pairs.append(extract_entity(i))
tqdm._instances.clear()

relations = [extract_relation(i) for i in tqdm(preprocessed_sentence
s)]
tqdm._instances.clear()

processed_entity_pairs = []
processed_relations = []
for i in tqdm(range(len(entity_pairs))):
  if entity_pairs[i][0]!='' and entity_pairs[i][1]!='':
    processed_entity_pairs.append(entity_pairs[i])
    processed_relations.append(relations[i])
tqdm._instances.clear()

source = []
target = []
edge = []
for i in tqdm(range(len(processed_entity_pairs))):
  doc_source = nlp(processed_entity_pairs[i][0]).ents
  str_source = [str(word) for word in doc_source]
  doc_source = ' '.join(str_source)
  doc_target = nlp(processed_entity_pairs[i][1]).ents
  str_target = [str(word) for word in doc_target]
  doc_target = ' '.join(str_target)
  if doc_source != '' or doc_target != '':
    edge.append(processed_relations[i])
    source.append(processed_entity_pairs[i][0])
    target.append(processed_entity_pairs[i][1])

plt.figure(figsize=(14,14))
```

```
df = pd.DataFrame({'source':source, 'target':target, 'edge': edge})
G = nx.from_pandas_edgelist(df=df[df['edge']=="directed"], source='s
ource', target='target', edge_attr='edge', create_using=nx.DiGraph()
)
pos = nx.spring_layout(G, k=0.5)
nx.draw(G, pos, with_labels=True, node_color='yellow', node_size=150
0, edge_cmap=plt.cm.Blues)
labels = {e: G.edges[e]['edge'] for e in G.edges}
nx.draw_networkx_edge_labels(G, pos, edge_labels=labels)
plt.show()
```

ДОДАТОК Б

Відомість кваліфікаційної роботи

| Позначення | Найменування | Дод. відомості |
|---|---|---|
| | Текстові документи | |
| **1.** | Пояснювальна записка | _103_ с. |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | Інші документи | |
| | | |
| **2.** | Презентаційні матеріали | __ плакатів |
| | | |
| | | |
| | | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | | | |
| | | | | | | | |
| . | . | Прізвище та ініціали. | Підп. | Дата | | | |
| Розробив | | Дмитрієв Д. Ю. | | | Методи подання знань засновані на обробці природно-мовних текстів та графів знань | Шифр групи | Код напр./спец. |
| Перевірив | | Рябова Н.В. | | | | СШІм-19-2 | 122 |
| Н.контр. | | Малєєва І.А. | | | | ХНУРЕ кафедра ШІ | |
| Затв. | | Філатов В.О. | | | | | |