

## ДОДАТОК А

### ЕЛЕМЕНТИ ПРОГРАМНОГО КОДУ

```

# ULMFiT + LSTM

df_dplr = df_dplr.reindex(np.random.permutation(df_dplr.index))
df_dplr.head()

import nltk
nltk.download('stopwords')

#Text Pre-processing
def text_cleaning(sentence):
    letters = re.sub("[^a-zA-Z]", " ", sentence)
    ht = re.sub(r'http\S+', '', letters)
    mention = re.sub(r'@\w+', '', ht)
    p = re.sub(r'^\w\s', '', mention)
    words = p.lower().split()
    stops = set(stopwords.words("english"))
    meaningful_words = [w for w in words if not w in stops]
    return(" ".join(meaningful_words))
df_dplr['text_clean'] = df_dplr['Phrase'].apply(lambda x:
text_cleaning(x))

Y = df_dplr['Sentiment']
lenc = LabelEncoder()
Y = lenc.fit_transform(Y)
Y = to_categorical(Y)

X_train, X_test, Y_train, Y_test =
train_test_split(df_dplr['text_clean'], Y, test_size=0.2, random_state=37)

max_words = len(set(" ".join(X_train).split()))
max_len = X_train.apply(lambda x: len(x)).max()
max_words, max_len

from keras.preprocessing.text import Tokenizer

#Tokenizer
tk = Tokenizer(num_words=max_words)
tk.fit_on_texts(X_train)
X_train_tk = tk.texts_to_sequences(X_train)
X_test_tk = tk.texts_to_sequences(X_test)

#Padding sequences
X_train_pad = sequence.pad_sequences(X_train_tk, maxlen=max_len)
X_test_pad = sequence.pad_sequences(X_test_tk, maxlen = max_len)

num_classes = 5
#Class weights Implementation for unbalanced dataset
def get_weight(y):

```

```

        class_weight_current = cw.compute_class_weight('balanced',
np.unique(y), y)
        return class_weight_current
    class_weight = get_weight(Y_train.flatten())
    print(class_weight)

def deep_lr_model(model):
    batch_size = 256
    epochs = 10
    optimizers = Adam(0.03)
    model.compile(optimizer='adam', loss='categorical_crossentropy',
metrics=['accuracy'])
    history = model.fit(X_train_pad, Y_train, epochs=epochs,
batch_size=batch_size, validation_data=(X_test_pad,Y_test),
verbose=1,class_weight=class_weight)
    return history

#base model
model = Sequential()
model.add(Embedding(max_words,128,input_length=max_len))
model.add(LSTM(128))
model.add(Dense(num_classes, activation='sigmoid'))
model.summary()

deep_lr_model(model)

#Adding dropout
model1 = Sequential()
model1.add(Embedding(max_words,128,input_length=max_len))
model1.add(LSTM(128))
model1.add(Dropout(0.4))
model1.add(Dense(num_classes, activation='sigmoid'))
model1.summary()

deep_lr_model(model1)

#Adding regularizer
model2 = Sequential()
model2.add(Embedding(max_words,128,input_length=max_len))
model2.add(Dropout(0.4))
model2.add(LSTM(128, kernel_regularizer=regularizers.l2(0.001)))
model2.add(Dense(num_classes, activation='sigmoid'))
model2.summary()

deep_lr_model(model2)

score = model.evaluate(X_test_pad,Y_test)
score1 = model1.evaluate(X_test_pad,Y_test)
score2 = model2.evaluate(X_test_pad,Y_test)

# XLNet

MODEL_CLASSES = {
    'bert': (BertForSequenceClassification, BertTokenizer,
BertConfig),

```

```

        'xlnet': (XLNetForSequenceClassification, XLNetTokenizer,
XLNetConfig),
    }

    # Parameters
    seed = 42
    use_fp16 = False
    bs = 16
    model_type = 'xlnet'
    pretrained_model_name = 'xlnet-base-cased'

    model_class, tokenizer_class, config_class =
MODEL_CLASSES[model_type]
    model_class.pretrained_model_archive_map.keys()

    transformer_tokenizer =
tokenizer_class.from_pretrained(pretrained_model_name)
    transformer_base_tokenizer =
TransformersBaseTokenizer(pretrained_tokenizer = transformer_tokenizer,
model_type = model_type)
    fastai_tokenizer = Tokenizer(tok_func = transformer_base_tokenizer,
pre_rules=[], post_rules=[])

    transformer_vocab = TransformersVocab(tokenizer =
transformer_tokenizer)
    numericalize_processor =
NumericalizeProcessor(vocab=transformer_vocab)

    tokenize_processor = TokenizeProcessor(tokenizer=fastai_tokenizer,
include_bos=False, include_eos=False)

    transformer_processor = [tokenize_processor, numericalize_processor]

    pad_first = bool(model_type in ['xlnet'])
    pad_idx = transformer_tokenizer.pad_token_id

    tokens = transformer_tokenizer.tokenize('Salut c est moi, Hello it s
me')
    print(tokens)
    ids = transformer_tokenizer.convert_tokens_to_ids(tokens)
    print(ids)
    transformer_tokenizer.convert_ids_to_tokens(ids)

    databunch = (TextList.from_df(train, cols='Phrase',
processor=transformer_processor)
        .split_by_rand_pct(0.1, seed=seed)
        .label_from_df(cols= 'Sentiment')
        .add_test(test)
        .databunch(bs=bs, pad_first=pad_first, pad_idx=pad_idx))

    config = config_class.from_pretrained(pretrained_model_name)
    config.num_labels = 5
    config.use_bfloat16 = use_fp16
    print(config)

```

```

    transformer_model =
model_class.from_pretrained(pretrained_model_name, config = config)
    custom_transformer_model = CustomTransformerModel(transformer_model =
transformer_model)

    from fastai.callbacks import *
    from transformers import AdamW
    from functools import partial

    CustomAdamW = partial(AdamW, correct_bias=False)

    learner = Learner(databunch,
                      custom_transformer_model,
                      opt_func = CustomAdamW,
                      metrics=[accuracy, error_rate])

    learner.callbacks.append(ShowGraph(learner))

    if use_fp16: learner = learner.to_fp16()

    learner.freeze_to(-1)
    learner.lr_find()

    learner.fit_one_cycle(1, max_lr=1e-04, moms=(0.8, 0.7))

    learner.freeze_to(-2)
    lr = 1e-5

    learner.fit_one_cycle(1, max_lr=slice(lr*0.95**num_groups, lr),
moms=(0.8, 0.9))

    learner.freeze_to(-3)
    learner.fit_one_cycle(1, max_lr=slice(lr*0.95**num_groups, lr),
moms=(0.8, 0.9))

    learner.unfreeze()
    learner.fit_one_cycle(2, max_lr=slice(lr*0.95**num_groups, lr),
moms=(0.8, 0.9))

    path = '/content/drive/My Drive/ML'
    export_learner = load_learner(path, file = 'xlnet_final.pkl')

```

ДОДАТОК Б  
**СЛАЙДИ ПРЕЗЕНТАЦІЇ**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Атестаційна робота магістра

Дослідження методів аналізу тональності тексту з використанням мереж-  
трансформерів

Виконав: ст. гр. ІПЗм-18-3

Науковий керівник: к.т.н., доц. каф. ПІ

Шуляк С.М.

Валенда Н.А.

2020

## Мета роботи

- Дослідження методів аналізу тональності тексту з використанням мереж-трансформерів: аналіз моделей ULMFiT, BERT та XLNet
- Експериментальне дослідження для виявлення з метою з'ясування їх придатності для класифікації відгуків споживачів інтернет магазинів

## Актуальність роботи

- Вибір правильної моделі класифікації відгуків споживачів інтернет-магазинів є одним з чинників, від яких залежить успішність маркетингової кампанії
- Пошук найбільш ефективного способу сентимент-аналізу є важливим і актуальним в умовах сучасного розвитку машинного навчання та маркетингу

## Постановка задачі

- Дослідження актуальних методів аналізу тональності тексту
- Дослідження методів обробки природних мов з використанням нейронних мереж-трансформерів
- Дослідження вирішення задачі класифікації сентиментів за допомогою методів з використанням трансформерів
- Проведення експерименту для виявлення моделі з найбільшим показником точності класифікації відгуків споживачів інтернет-магазинів

## Методи аналізу тональності тексту

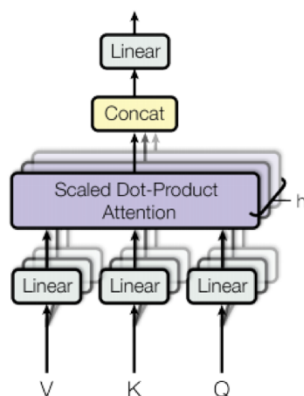
- Методи, засновані на правилах і словниках
- Методи машинного навчання з учителем
- Методи машинного навчання без учителя
- Метод з використанням теоретико-графових моделей

2020

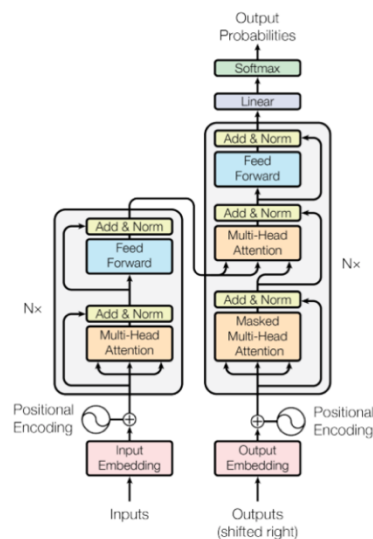
5

## Методи обробки з використанням мереж-трансформерів

- BERT - Bidirectional Encoder Representations from Transformers
- XLNet



2020

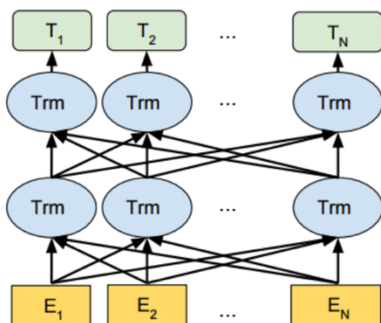


6

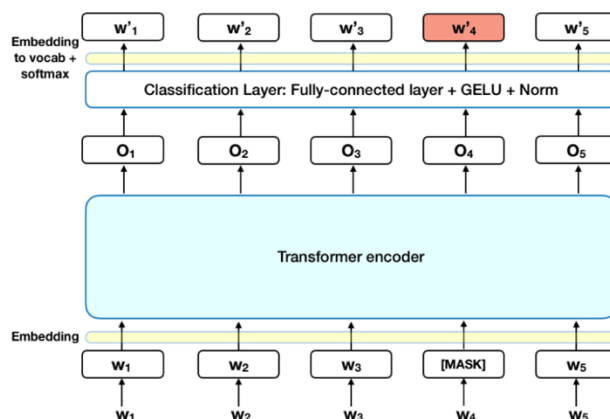
## BERT

- Найкращий показник Ассурасу: 95.79

- Тип: автоенкодер



- Запропонований розмір мережі:  $L = 12$ ,  $H = 768$ ,  $A = 12$   
кількість параметрів = 110 мільйонів



2020

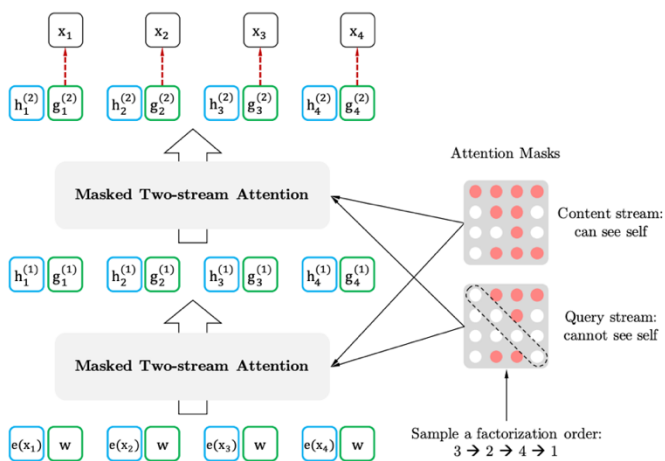
7

## XLNet

- Найкращий показник

Ассурасу - 96.21

- Тип: авторегресія



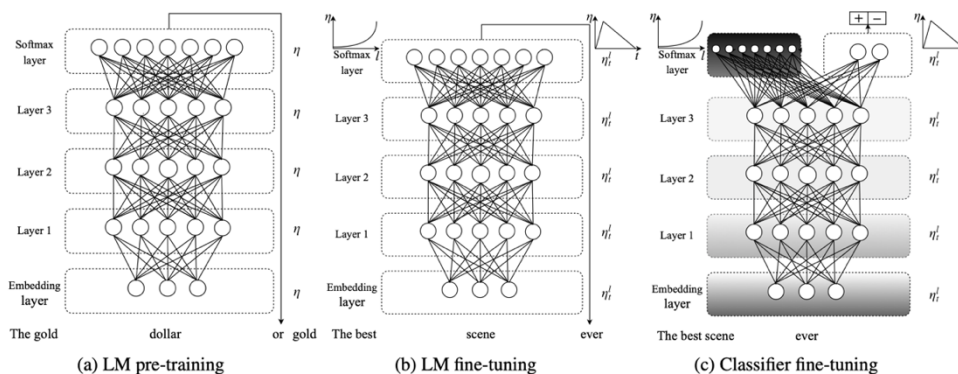
2020

8

## ULMFiT

- Метод тонкої настройки мовних моделей
- Похилий трикутний коефіцієнту навчання
- Поступове розмороження шарів моделі

Дозволяє зменшити похибку на 18-24%

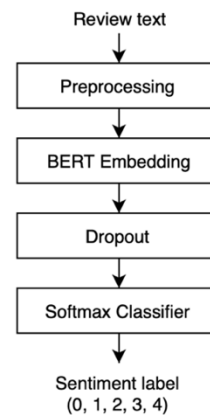


2020

9

## Використання BERT для сентимент-аналізу

- BERT з тонкою настройкою на датасеті SST має точність 94.7 (5 класів)

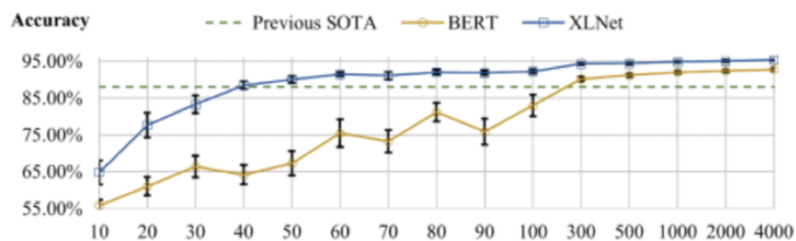


2020

10

## Використання XLNet для сентимент-аналізу

- XLNet з тонкою настройкою має точність 95.13 (5 класів)

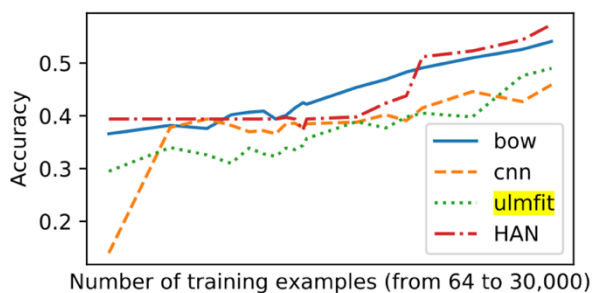


2020

11

## Використання ULMFiT для сентимент-аналізу

- ULMFiT з тонкою настройкою має точність 69.4 (5 класів)

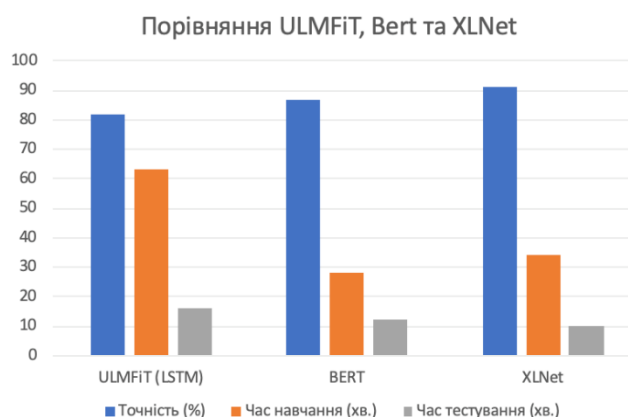


2020

12

## Експериментальне порівняння моделей

- Було розроблено моделі на базі BERT, XLNet та LSTM з ULMFiT
- Моделі натреновано на наборі даних Amazon Product Review
- Класифікація за 5 класами: «дуже негативно» – «дуже позитивно»



2020

13

## Висновки

- Досліджено актуальні методи аналізу тональності тексту
- Досліджено методи обробки природних мов з використанням нейронних мереж-трансформерів
- Досліджено вирішення задачі класифікації сентиментів за допомогою методів з використанням трансформерів
- Проведено експерименти для виявлення моделі з найбільшим показником точності класифікації відгуків споживачів інтернет-магазинів

Шуляк С.М. 2020

14

ДОДАТОК В  
АПРОБАЦІЯ РЕЗУЛЬТАТІВ РОБОТИ

**TOPICAL ISSUES OF THE  
DEVELOPMENT OF  
MODERN SCIENCE**

Abstracts of IX International Scientific and Practical Conference

Sofia, Bulgaria

6-8 May 2020

**Sofia, Bulgaria**

**2020**

**UDC 001.1**

**BBK 91**

The 9<sup>th</sup> International scientific and practical conference “Topical issues of the development of modern science” (May 6-8, 2020) Publishing House “ACCENT”, Sofia, Bulgaria. 2020. 968 p.

**ISBN 978-619-93537-5-2**

The recommended citation for this publication is:

*Ivanov I. Analysis of the phaunistic composition of Ukraine // Topical issues of the development of modern science. Abstracts of the 9th International scientific and practical conference. Publishing House “ACCENT”. Sofia, Bulgaria. 2020. Pp. 21-27. URL: <http://sci-conf.com.ua>.*

**Editor**

**Komarytskyy M.L.**

*Ph.D. in Economics, Associate Professor*

**Editorial board**

Dessislava Iosifova, VUZF University, Bulgaria  
Aleksander Aristovnik, University of Ljubljana, Slovenia  
Efsthathios Dimitriadi, Kavala Institute of Technology, Greece  
Eva Borszeki, Szent Istvan University, Hungary  
Fran Galetic, University of Zagreb, Croatia  
Goran Kutnjak, University of Rijeka, Croatia  
Janusz Lyko, Wroclaw University of Economics, Poland  
Ljerka Cerovic, University of Rijeka, Croatia

Ivane Javakhishvili Tbilisi State University, Georgia  
Marian Siminica, University of Craiova, Romania  
Mirela Cristea, University of Craiova, Romania  
Olga Zaborovskaya, State Institute of Economics, Russia  
Peter Joehnk, Helmholtz - Zentrum Dresden, Germany  
Zhelio Hristozov, VUZF University, Bulgaria  
Toma Sorin, University of Bucharest, Romania  
Velizar Pavlov, University of Ruse, Bulgaria  
Vladan Holcner, University of Defence, Czech Republic

Collection of scientific articles published is the scientific and practical publication, which contains scientific articles of students, graduate students, Candidates and Doctors of Sciences, research workers and practitioners from Europe, Ukraine, Russia and from neighbouring countries and beyond. The articles contain the study, reflecting the processes and changes in the structure of modern science. The collection of scientific articles is for students, postgraduate students, doctoral candidates, teachers, researchers, practitioners and people interested in the trends of modern science development.

**e-mail:** [sofia@sci-conf.com.ua](mailto:sofia@sci-conf.com.ua)

**homepage:** <http://sci-conf.com.ua>

©2020 Scientific Publishing Center “Sci-conf.com.ua” ®

©2020 Publishing House “ACCENT” ®

©2020 Authors of the articles

- |      |   |     |
|------|---|-----|
| 127. | <b>Черная Т. В., Гасай Г. А.</b><br>КОМПЛЕКСНАЯ ИГРОВАЯ КИНЕЗИТЕРАПИЯ В<br>РЕАБИЛИТАЦИИ ДЕТЕЙ С АУТИЗМОМ.   | 876 |
| 128. | <b>Чернова К. В., Чорна Ю. В.</b><br>МУЛЬТИМЕДІЙНІ ПРЕЗЕНТАЦІЇ ЯК ЗАСІБ ІНТЕНСИФІКАЦІЇ<br>ПРОЦЕСУ НАВЧАННЯ ІНОЗЕМНИХ СЛУХАЧІВ НА<br>ДОВУЗІВСЬКОМУ ЕТАПІ.  | 882 |
| 129. | <b>Чорна Ю. В., Янушевська І. Б.</b><br>КОНТРОЛЬ НА ДОВУЗІВСЬКОМУ ЕТАПІ НАВЧАННЯ (З<br>ДОСВІДУ РОБОТИ ЦЕНТРУ ПІДГОТОВКИ ІНОЗЕМНИХ<br>ГРОМАДЯН ЗДМУ).      | 887 |
| 130. | <b>Шапакидзе Е. Д., Николеишвили Г. В.</b><br>РОЛЬ ИННОВАЦИОННЫХ ТЕХНОЛОГИИ В РАЗВИТИИ<br>ШЕЛКОВОДСТВА ГРУЗИИ.  | 894 |
| 131. | <b>Шевців Л. Ю., Сенишин Б. Б.</b><br>ШЛЯХИ ПІДВИЩЕННЯ ФІНАНСОВОЇ БЕЗПЕКИ БАНКІВ<br>УКРАЇНИ В УМОВАХ ГЛОБАЛІЗАЦІЙНИХ ПРОЦЕСІВ.                            | 903 |
| 132. | <b>Шевців Л. Ю., Сподарик Т. І.</b><br>СТАН І ПЕРСПЕКТИВИ РОЗВИТКУ БАНКІВСЬКОЇ СИСТЕМИ<br>УКРАЇНИ.  | 911 |
| 133. | <b>Шевчук І. В., Бурлака Л. П.</b><br>МЕТОДИЧНІ ПІДХОДИ ДО ВИВЧЕННЯ ТАБЛИЧНИХ<br>ВИПАДКІВ МНОЖЕННЯ ТА ДІЛЕННЯ НА УРОКАХ<br>МАТЕМАТИКИ В ПОЧАТКОВІЙ ШКОЛІ. | 919 |
| 134. | <b>Шевчук О. А., Вергеліс В. І.</b><br>НАСІННЄВА ПРОДУКТИВНІСТЬ СОЇ ЗА ВИКОРИСТАННЯ<br>РІСТРЕГУЛЮЮЧИХ ПРЕПАРАТІВ.   | 928 |
| 135. | <b>Шомиров Д. А.</b><br>ТОТАЛЬНАЯ ГИСТЕРЭКТОМИЯ У ЖЕНЩИН С ОЖИРЕНИЕМ.   | 938 |
| 136. | <b>Шмуліч О. В., Алиева С. В.</b><br>ПРОГНОСТИЧНА ДИНАМІКА В КАРДІАЛЬНІЙ<br>ТРАНСПЛАНТАЛОГІЇ.   | 947 |
| 137. | <b>Шуляк С. М., Валенда Н. А.</b><br>ЗАДАЧА АНАЛІЗУ ТОНАЛЬНОСТІ ТЕКСТУ.   | 951 |
| 138. | <b>Юркова Т. Ф.</b><br>ФОРМУВАННЯ В УЧНІВ ПІДЛІТКОВОГО ВІКУ ЦІННІСНОГО<br>СТАВЛЕННЯ ДО ПРИРОДИ ЗАСОБАМИ УКРАЇНСЬКОЇ<br>ЛІТЕРАТУРИ.                        | 957 |
| 139. | <b>Яровицька Н. А., Кулібаба О. С.</b><br>АКТУАЛЬНІСТЬ ІДЕЙ АЛЬБЕРА КАМЮ В УМОВАХ<br>ЗАГАЛЬНОСВІТОВОГО КАРАНТИНУ.   | 966 |

УДК 004.89

## ЗАДАЧА АНАЛІЗУ ТОНАЛЬНОСТІ ТЕКСТУ

**Шуляк Сергій Михайлович**

студент

Харківський національний університет радіоелектроніки

Україна, м. Харків

**Валенда Наталія Анатоліївна**

к.т.н., доцент

Харківський національний університет радіоелектроніки

Україна, м. Харків

**Анотація:** стаття присвячена огляду задачі аналізу тональності тексту, що є однією з задач обробки природної мови, а також методу для її вирішення за допомогою попередньо тренованої моделі з використанням нейронної мережі-трансформера.

**Ключові слова:** обробка природної мови, аналіз тональності тексту, нейронні мережі, мережі-трансформери

Задача аналізу тональності тексту пов'язана з широким спектром різноманітних завдань, де використовуються текстові дані, наприклад: аналіз думки споживачів при маркетингових дослідженнях, аналіз відгуків споживачів про певний продукт, пошук трендів, аналіз загального сприйняття продукту споживачами, виявлення важливих аспектів продукту, тощо. Ця задача є однією з задач обробки природної мови.

Обробка природної мови (англ. *Natural-language processing, NLP*) – це розділ лінгвістики, інформатики та штучного інтелекту, що стосується взаємодії між комп'ютерами та людськими (природними) мовами. Вивчає обробку та аналіз великої кількості даних природної мови за допомогою комп'ютера [1, с. 32].

Аналіз тональності тексту полягає у пошуку та аналізі емоційного забарвлення тексту та ставлення автора тексту до певного об'єкту. Емоційна складова певного фрагменту тексту, наприклад речення, називається лексичним сентиментом (або лексичною тональністю). Тональність всього тексту можна представити як функцію від лексичних сентиментів та правил їх поєднання.

Складність аналізу тональності тексту полягає в тому, що існують випадки, коли слова в реченні поєднані таким чином, що тональність не відповідає словам за допомогою яких вона передається. Наприклад: «Я люблю свій телефон, але не порекомендую його будь-кому з моїх колег». У цьому реченні присутні як позитивна так і негативна частина, тому важко класифікувати таке речення як позитивне чи негативне.

Задача аналізу тональності по суті є задачі класифікації, оскільки після виявлення ознак в лексемах і співставлення за певними правилами, текст необхідно класифікувати.

**Класифікація тексту при сентимент-аналізі зазвичай поділяється на:**

- за бінарною шкалою;
- за багатосмуговою шкалою;
- за системами шкалювання (наприклад від -10 до 10);
- суб'єктивність / об'єктивність тексту.

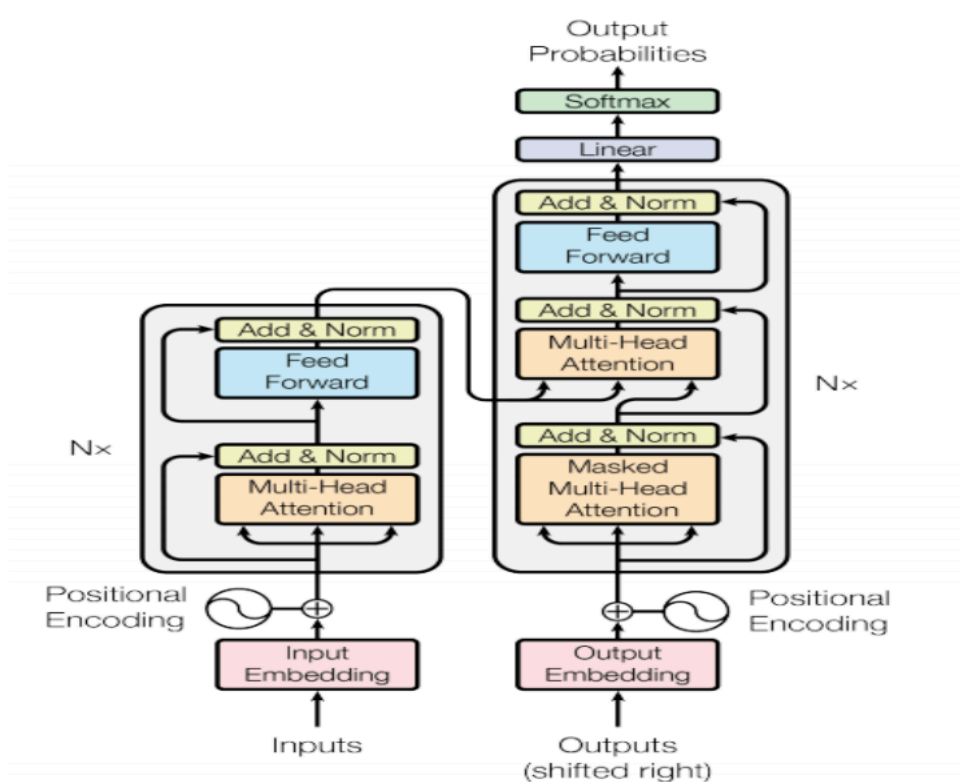
**Вирішення задачі аналізу тональності тексту може проводитися різними методами:**

- моделі, засновані на правилах та словниках;
- машинне навчання з вчителем
- машинне навчання без вчителя
- метод теоретико-графових моделей

Найсучаснішим методом вирішення різних задач обробки природних мов є використання попередньо тренуваних моделей та нейронних мереж з архітектурою «трансформер» (див. рис. 1).

Найвідоміший приклад такої моделі – BERT (Bidirectional Encoder Representations from Transformers) від Google. Її ідея полягає в тому, що на вхід

нейронної мережі-трансформера подається текст, де 15% слів замінено на спеціальний токен-маску, який на виході нейронної мережі повинен бути замінений на слово, що підходить по контексту. Також нейронна мережа має робити висновок якщо речення, що обробляється є логічним продовженням попереднього речення. Якщо через такий процес навчання пропустити великий обсяг текстів, наприклад, Wikipedia, то буде отримана дуже якісна модель, яку можна використовувати для різноманітних задач з мінімальними змінами.

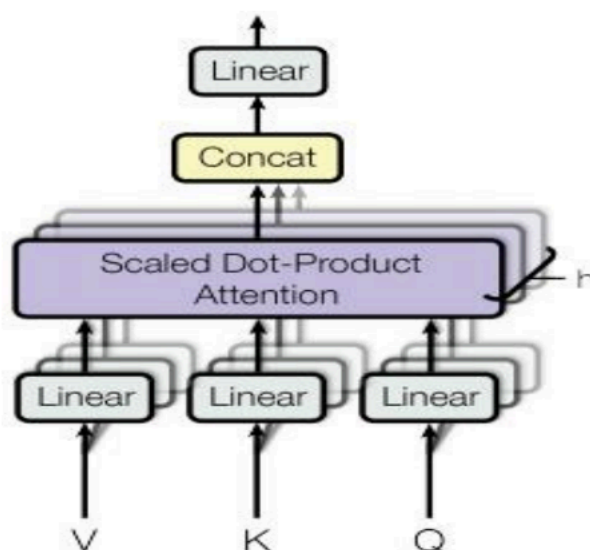


**Рис. 1. Архітектура мережі-трансформера**

Мережа в такому вигляді складається з енкодера з шаром багатоголосової уваги (multi-head attention) та декодера.

Енкодер отримує на вхід слова і видає метаданні, що відповідають цим словам для подальшого використання у декодері. Кожне слово паралельно проходить через шари енкодера, де части повнозв'язних шарів (fully-connected layers), а деякі – скорочені (shortcut connections).

Елементом, який відрізняє цю архітектуру від попередніх є шар багатоголової уваги (multi-head attention). Multi-head attention. Цей шар дає можливість кожному вхідному вектору взаємодіяти з іншими через механізм уваги (attention mechanism), а не використовувати схований стан (hidden state), як це відбувається в RNN (рекурентних нейронних мережах) або сусідніх слів в CNN (згорткових нейронних мережах). Структура цього шару наведена на рисунку 2.



**Рис. 2. Структура шару multi-head attention**

На вхід шару подаються вектори  $Q$  і декілька пар  $K$  і  $V$ . Зазвичай  $K$  і  $V$  це один і той же вектор. Кожен з них перетворюється лінійним перетворенням зі здатністю навчатися, обчислюється скалярний добуток  $Q$  з усіма  $K$  по черзі. Результат скалярних добутків через функцію softmax. Отримуються ваги, з якими усі вектори  $V$  сумуються в один вектор. Векторів уваги (attention) одночасно тренується декілька ( $h$ ), після чого результат конкатенується і проходить через лінійне перетворення і передається далі у енкодері. Завдяки тому що на виході з цього блоку отримуються вектори однакового розміру їх можна використати в мережі декілька разів.

Однією з ознак кожного слова є позиційне кодування (positional encoding), завдяки чому є здатність «звертати увагу» на його контекст – сусідні слова в реченні.

Декодер запускається на одне слово, отримує на вхід попереднє слово і повинен видаляє наступне.

**У декодера є два типи використання багатоголосової уваги (multi-head attention):**

- звернення до векторів минулих декодованих слів;
- звернення до виходу енкодера. В цьому випадку Q – це вхідний вектор декодеру, а пари K-V – це фінальні метаданні енкодера, де один і той же вектор йде в якості і K, і V, але проходять через різні перетворення.

Процес повторюється кілька разів, де результат одного блоку передається наступному. Над результатом виконується функція softmax для отримання ймовірностей слів. Після цього виконується семплювання результатом якого буде наступне слово у реченні. Результат передається на вхід декодеру поки не закінчиться речення [2].

**Переваги використання трансформерів:**

- можливість роботи з дуже великими обсягами даних для навчання;
- можливість розпаралелювання;
- самостійне навчання;
- можливість трансферного навчання (відсутня необхідність створення власної моделі з нуля);
- високий показник точності (accuracy), близький до 90%.

**Недоліки:**

- тренування моделі з нуля є дуже затратним і неможливе на користувацькому обладнанні та більшості серверів.

Завдяки здатності цього підходу до трансферного навчання, зазвичай використовують BERT-модель в якості бази для нейронної мережі і додають ще декілька шарів для адаптації всієї мережі до певної конкретної задачі, наприклад для аналізу тональності.

Таким чином, на даний момент використання попередньо-тренованих моделей є дуже перспективним методом вирішення різноманітних задач обробки природних мов, зокрема аналізу тональності тексту.

### СПИСОК ЛІТЕРАТУРИ

1. Liu, Bing (2010). "Sentiment Analysis and Subjectivity" (PDF). In Indurkha, N.; Damerau, F. J. (eds.). Handbook of Natural Language Processing (Second ed.).
2. Open Sourcing BERT: State-of-the-Art Pre-training for Natural Language Processing // Google AI Blog. URL: <https://ai.googleblog.com/2018/11/open-sourcing-bert-state-of-art-pre.html> (дата звернення: 22.03.2020).

**ДОДАТОК Г**  
**ВІДГУК НА АТЕСТАЦІЙНУ РОБОТУ МАГІСТРА**

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет комп'ютерних наук

**ВІДГУК**

на атестаційну роботу магістра  
**Шуляка Сергія Михайловича, ІПЗм-18-3**

спеціальність *121- Інженерія програмного забезпечення*  
Освітньо-наукова програма *«Інженерія програмного забезпечення»*

Тема атестаційної роботи «Дослідження методів аналізу тональності тексту з використанням мереж-трансформерів»

Представлена атестаційна робота магістра відповідає затвердженій темі та виконана відповідно до завдання.

В роботі були проведені дослідження методів аналізу тональності тексту з використанням мереж-трансформерів. Методи дослідження полягають у теоретичному дослідженні існуючих рішень задачі багатокласової класифікації сентиментів у тексті. Проведено експериментальне порівняння моделей ULMFiT, BERT та XLNet на відгуках споживачів про товари. В результаті роботи було виявлено найбільш придатну модель для вирішення вищевказаної задачі за показником точності.

За час атестаційної роботи Шуляк С.М. показав вміння розбиратися з новою інформацією, швидко на її основі пропонувати нові рішення, виявив гарні навички в опрацюванні технічної літератури та документації. При реалізації задачі використовував кращі практики вибраної технології. Це дозволило йому швидко та грамотно реалізувати поставлену задачу. Роботу магістра можна назвати ефективною та наперед спланованою.

Завдання були виконані без відхилень від календарного плану. Роботу було виконано своєчасно та у відповідності до поставленої задачі. Пояснювальна записка виконана згідно вимогам.

Магістрант гр. ІПЗм-18-3 Шуляк С.М. готов до самостійної інженерної діяльності.

Атестаційну роботу можна подати до захисту в ЕК за спеціальністю 121 - «Інженерія програмного забезпечення», освітньо-науковою програмою Інженерія програмного забезпечення.

«18» 05. 2020р.

Керівник атестаційної роботи магістра  
к.т.н, доц. Валенда Н.А.