

УДК 004.912

1.11. Эмоциональный анализ взаимодействия участников информационной системы

Коваленко Михаил Андреевич, РТУ МИРЭА, Москва, Россия
Исаева Аделина Владимировна, РТУ МИРЭА, Москва, Россия

В статье рассматривается проблем оценка текстовой информации в информационных системах с акцентом на доверие. Авторы подчеркивают важность формирования доверия по различным критериям, таким как авторитетность и экспертность авторов, надежность источников, ясность изложения и соответствие информации потребностям. В статье представлена формализация эмерджентности доверия в виде кода на python.

Введение

С ростом цифровизации возрастает актуальность оценки информационных систем с точки зрения надежности как совокупности факторов, определяющих достоверность информации в рамках этой системы. Доверие в данном контексте может формироваться по многим причинам, включая авторитетность и экспертность авторов, надежность и достоверность источников, четкость и ясность изложения, а также соответствие информации нашим потребностям и интересам.

Эмерджентность доверия - это процесс, связанный напрямую с ростом объема информации и улучшением ее качества, а также с увеличением числа проверок и подтверждений, основанных на отзывах, рейтингах и других значимых индикаторах, в ходе которого доверие к представленной информации, укрепляется и развивается [Коваленко, 2021].

Определение доверия в информационной среде может включать в себя анализ текста и выявление сущностей, связей и других факторов, которые могут влиять на доверие между участниками, включая в себя авторитет авторов, надежность и достоверность источников, ясность и понятность изложения.

Анализ эмоциональной окраски текстов представляет собой важную цель в исследовании эмоций. Поскольку слова и выражения, используемые в текстах, часто нестандартны и определение эмоциональной полярности текстов представляет собой сложную задачу. Человеческие эмоции охватывают разнообразные аспекты, включая поведенческие, когнитивные, психологические и физиологические особенности, и исследование и анализ человеческих эмоций и чувств имеют большое значение в области когнитивных вычислений [Райков, 2022].

Внутренняя взаимосвязь между данными признаками была недостаточно исследована, но она может оказаться ключевым фактором для повышения понимания работы нейронных сетей.

Исследования последних трех десятилетий показали, что эмоции потребителей существенно влияют на их суждения и принятие решений [Han S., 2007]. Сейчас происходит формирование научно-технологической базы для разработки эффективных инструментов функционирования интеллектуальных систем [Турдаков, 2022]. Одним из инструментов является заслуживающий доверия искусственный интеллект [ГОСТ Р 59 276–2020]. Доверие к системам искусственного интеллекта является важнейшим условием, определяющим возможность применения этих систем при решении ответственных задач обработки данных.

Один из важных аспектов языка заключается в способности текста передавать эмоциональное состояние автора и воздействовать на чувства читателя. С развитием нейронных сетей появилась возможность создания текстов с выраженной эмоциональной окраской, что, в некоторых случаях, может быть полезным [Козырев, 2019]. Тем не менее, важно оценивать качество сгенерированных текстов, включая их содержание. В ходе анализа эмоциональной окраски текста можно прийти к определению его общего настроения (например, "положительное", "отрицательное" и "нейтральное"), а также выявить конкретные эмоциональные состояния. Анализ эмоциональной окраски текста может осуществляться путем исследования лингвистических признаков, которые способны отражать тональность и смысловой оттенок текста.

Развитие нейронных сетей предоставило возможность создавать тексты, обладающие выраженной эмоциональной окраской [Райков, 2022]. Это может оказаться полезным, поскольку способность текста воздействовать на чувства читателя и передавать эмоциональное состояние автора играет важную роль в формировании доверия.

Программная реализация

Описание приведенных выше моделей позволит реализовать программное представление данной системы на языке Python. Модель состоит из 3 основных компонентов:

- Агент, определяемый репутацией $\in [0,1]$, где 0 – минимальное значение репутация, 1 – максимальное;

- Взаимодействие от агента i к агенту j , определяемое эмоциональной окраской $\in [-1,1]$ и объективность $\in [0,1]$;
- Доверие, определяемое численно от $[0,1]$.

Принцип работы системы

Пусть существует агент i с репутацией $reputation_i$ и агент J , число доверие от агента j к агенту i – $trust_{ji}$, а каждое сообщение между агентами будет определяться как взаимодействие с эмоциональной окраской $sentiment$ и объективностью $objectivity$, тогда доверие от агента j к агенту i изменяется следующим образом:

$$trust_{ji} = trust_{ji} + \Delta trust_{ji},$$

где $\Delta trust_{ji} = reputation_i * objectivity * sentiment * k$,

где k – коэффициент от 0 до 1.

```
def delta_trust(self):
    trust_change = k * self.get_sentiment() * self.get_interaction_type() * self._agent1.reputation
    trust = self._agent2.get_trust_score_by_id(agent_id=self._agent1.ID)
    if 0 < trust.get_score() + trust_change < 1:
        trust.set_score(trust.get_score() + trust_change)
```

Эмоциональная окраска и субъективность сообщения определяется с помощью библиотеки Dostoevsky, которая оценивает нейтральность, негативность и позитивность.

Вот как выполнить эмоциональный анализ слов с использованием Dostoevsky:

1. Установите библиотеку Dostoevsky, используя ReadMe;
2. Импортируйте библиотеку и проведите анализ сентимента:

```
from dostoevsky.tokenization import RegexTokenizer
from dostoevsky.models import FastTextSocialNetworkModel

def analyze_sentiment(word):
    tokenizer = RegexTokenizer()
    tokens = tokenizer.split('все очень плохо') #
    model = FastTextSocialNetworkModel(tokenizer=tokenizer)
    model.predict(review, k=6)
    subjectivity = float(result[0]['neutral'])
    polarity = float(result[0]['positive']) - float(result[0]['negative'])
    return polarity, subjectivity

word = "все очень плохо"
polarity, subjectivity = analyze_sentiment(word)

print(f"Word: {word}")
print(f"Polarity: {polarity}")
print(f"Subjectivity: {subjectivity}")
```

Значение полярности варьируется от -1 (отрицательное) до 1 (положительное), а значение субъективности от 0 (объективное) до 1 (субъективное). В этом примере мы анализируем эмоциональную окраску слова "все очень плохо", но вы можете использовать этот код для анализа любых других слов или текстов.

Также стоит отметить, что Dostoevsky основан на правилах и предварительно обученных моделях, и его точность может быть ограничена. Для более точного анализа сентимента можно использовать другие инструменты и подходы, например обучение собственной модели на основе набора данных с аннотацией сентимента.

Теперь, когда получено доверие между агентами, мы сможем вычислить эмергентность доверия между этими агентами: оно зависит от вычисленного ранее доверия, от количества взаимодействий от j к i и от общего количества взаимодействий в системе.

$EmergentTrust(i,j) = (InteractionCount(i,j) * TrustScore(i,j)) / \text{Sum}(InteractionCount(x,y) * TrustScore(x,y))$, где:

- $EmergentTrust(i,j)$ - эмергентность доверия между пользователями i и j
- $InteractionCount(i,j)$ - количество взаимодействий между пользователями i и j
- $TrustScore(i,j)$ - оценка доверия между пользователями i и j

• $\text{Sum}(InteractionCount(x,y) * TrustScore(x,y))$ - сумма произведений количества взаимодействий и оценок доверия для всех пар пользователей в системе.

Ниже представленная программная реализация на языке Python.

```

• # Calculate the emergent trust score for agent_i and agent_j based on their interactions with all agents
@staticmethod
def calculate_for_i_j(agent_i, agent_j, agents):
    trust_score = agent_i.get_trust_score_by_id(agent_id=agent_j.ID)
    if trust_score:
        trust_score = trust_score.get_score()
    else:
        trust_score = 0
    interaction_count_ij = agent_i.get_interactions_by_id_count(agent_id=agent_j.ID)
    sum = 0
    for x in agents:
        for y in agents:
            if not x.ID == y.ID:
                if not x.get_interactions_by_id_count(agent_id=y.ID) == 0:
                    sum += x.get_interactions_by_id_count(agent_id=y.ID) * \
                        y.get_trust_score_by_id(agent_id=x.ID).get_score()
    return trust_score * interaction_count_ij / sum

```

Тогда для вычисления средней эмергентности доверия в системе необходимо вычислить среднее арифметическое для всех эмергентностей доверия:

```

# Calculate the average emergent trust score for all agents in the system
@staticmethod
def calculate_average(agents):
    sum, count = 0, 0
    for a1 in agents:
        for a2 in agents:
            if not a1 == a2:
                sum += EmergentTrust.calculate_for_i_j(a1, a2, agents)
                count += 1
    return sum / count

```

Взаимодействию в системе может быть отображено двумя способами:

- с помощью графа,
- с помощью тепловой карты.

Агентная модель эмерджентности доверия на основе графов

Эмерджентность доверия может быть описана как возникновение доверия из составляющих системы, например, в отношениях между людьми или группами. Создание математической модели для таких сложных и абстрактных концепций, как доверие, требует определения их ключевых компонентов и изучения взаимодействия между ними.

В социальных сетях и организационных структурах эмерджентность доверия может возникать из локальных взаимодействий и влиять на глобальные свойства системы, такие как устойчивость, кооперация и информационный обмен.

Примечательно, что математическое моделирование эмерджентности доверия является сложной задачей, поскольку доверие и связанные с ним факторы могут быть очень абстрактными и зависеть от контекста.

Одна из возможных моделей, которую можно рассмотреть, – это агентная модель на основе графов. Демонстрация посредством графа включает в себя следующее: граф состоит из вершин, которые представляют собой агентов, а именно их репутацию. Цвет вершины отображает уровень репутации: красный – плохая репутация, зеленый – хорошая.

Граф ориентирован. Каждое ребро представляет собой доверие между агентами, вес ребра определяет его численное значение, цвет отображает уровень доверия по аналогии с цветом вершины, а толщина ребра – количество взаимодействий.

Для анализа эмерджентности доверия можно использовать для создания, анализа и визуализации сложных сетей библиотеки NetworkX и Matplotlib. В данном примере описана функция демонстрации системы с помощью графа.

```

# Function to draw a graph visualizing the trust relationships between agents
def draw_graph(agents, name_of_file):
    # Create a directed graph (digraph) and add nodes for each agent
    graph = nx.MultiDiGraph()
    graph.add_nodes_from(agents)

```

```

# Add trust relationships (edges) between agents based on trust scores
trusts = []
for v in agents:
    for one in v.get_trust_scores():
        trusts.append(one)
for trust in trusts:
    try:
        graph.add_edge(trust.get_agent1(), trust.get_agent2(),
                       trust=EmergentTrust.calculate_for_i_j(trust.get_agent1(), trust.get_agent2(), agents))
    except Exception:
        graph.add_edge(trust.get_agent1(), trust.get_agent2(),
                       trust=0.5)

# Find the maximum interaction count for scaling edge widths
max_int = max([u.get_interactions_by_id_count(v.ID) for u, v in graph.edges()])
if max_int == 0:
    max_int = 10

# Define the positions of nodes in a circular layout
pos = nx.circular_layout(graph)

# Set edge widths and colors based on interaction count and trust scores
edge_widths = [5 * (u.get_interactions_by_id_count(v.ID) + 1) / max_int for u, v in graph.edges()]
edge_colors = [
    (sqrt(1 - graph[u][v][0]['trust']), sqrt(graph[u][v][0]['trust']), 0) for u, v in graph.edges()
]

# Set node colors based on reputation scores
node_colors = [(sqrt(1 - v.get_reputation()), sqrt(v.get_reputation()), 0) for v in graph.nodes()]

# Draw the graph using networkx and matplotlib
nx.draw(
    graph,
    pos,
    node_size=500,
    node_color=node_colors,
    with_labels=True,
    edge_color=edge_colors,
    width=edge_widths,
    arrows=True,
    connectionstyle='arc3, rad = 0.1',
)
try:
    plt.title(f"AET for sys{EmergentTrust.calculate_average(agents)}")
except Exception:
    plt.title(f"AET for sys")
# Save the graph visualization as an image
plt.savefig("output/graph" + name_of_file)
plt.close()

```

Демонстрация системы через граф информативна для небольших систем.

Агентная модель эмерджентности доверия на основе тепловой карты

Следующий способ демонстрации - тепловая карта.

Для анализа эмерджентности доверия можно использовать для создания, анализа тепловых карт библиотеки seaborn и Matplotlib. На тепловой карте численные величины доверия между агентами представлены в виде матрицы, цвет ячейки отображает уровень доверия.

```

# Function to draw a heatmap based on trust scores between agents
def draw_heatmap(agents, name_of_file):
    # Prepare the data for the heatmap based on trust scores
    data = []
    for agent1 in agents:
        data1 = []
        for agent2 in agents:

```

```

data2 = agent1.get_trust_score_by_id(agent2.ID)
if agent1 == agent2:
    data1.append(-1) # Placeholder value for the diagonal (self-trust)
elif data2:
    try:
        data1.append(EmergentTrust.calculate_for_i_j(agent1, agent2, agents))
    except Exception:
        data1.append(0.5)
else:
    data1.append(0.5) # Default value for missing trust scores
data.append(data1)
data = np.array(data)

# Create the heatmap using seaborn and custom color map
hm = sns.heatmap(
    data=data,
    annot=True,
    mask=data == -1, # Hide the diagonal values (self-trust)
    cmap=LinearSegmentedColormap.from_list('red_green', ['r', 'y', 'g'], 256),
)

# Tweak the appearance of the plot
plt.tick_params(
    axis='both',
    which='major',
    labelsize=10,
    labelbottom=False,
    bottom=False,
    top=False,
    labeltop=True,
)
# Save the heatmap as an image

plt.savefig("output/heatmap" + name_of_file)
plt.close()

```

Выводы

Мы полагаем, что концепция эмерджентности доверия критически важна в контексте информационных систем. Это представляет собой взгляд на формирование доверия, где доверие формализуется из множества параметров взаимодействия и компонентов системы.

Отдельно можно выделить роль эмоциональной окраски текстов в формировании доверия, особенно в условиях развития нейронных сетей. Это представляет новые возможности и вызовы, так как эмоциональное воздействие текстов может влиять на восприятие и доверие пользователей.

Наши результаты могут быть использованы в дальнейших исследованиях, направленных на более глубокое понимание взаимосвязей в эмерджентности доверия, а также на разработку эффективных методов математического моделирования.

Литература

1. ГОСТ Р 59 276–2020 «Системы искусственного интеллекта. Способы обеспечения доверия. Общие положения»
2. Агеев А. И. и др. Социогуманитарные аспекты цифровых трансформаций и искусственного интеллекта: монография. М.: Общество с ограниченной ответственностью "Когито-Центр", 2022. 308 с.
3. Kovalenko, M. A., R. G. Bolbakov, and V. A. Mordvinov. "Analysis of the emergence of trust in the information field as a decision-making tool." *IFAC-PapersOnLine* 54.13 (2021): 183-187.
4. Козырев, А. Н. Социальный кредит в Китае: Обзор / А. Н. Козырев, И. В. Неволин // *Цифровая экономика*. – 2019. – № 3(7). – С. 70-74.
5. Турдаков Д. Ю. и др. Доверенный искусственный интеллект: вызовы и перспективные решения // *Доклады Российской академии наук. Математика, информатика, процессы управления*. – 2022. – Т. 508. – №. 0. – С. 13-18.
6. Райков А.Н. Субъектность объяснимого искусственного интеллекта // *Философские науки*. 2022. Т. 65. № 1. С. 72-90.
7. Han S., Lerner J. S., Keltner D. Feelings and consumer decision making: The appraisal-tendency framework // *Journal of consumer psychology*. – 2007. – Т. 17. – №. 3. – С. 158-168.

References in Cyrillics

1. ГОСТ Р 59 276–2020 «Системы искусственного интеллекта. Способы обеспечения доверия. Общие положения»
2. Агеев А. И. и др. Социогуманитарные аспекты цифровых трансформаций и искусственного интеллекта: монография. М.: Общество с ограниченной ответственностью "Когито-Центр", 2022. 308 с.
3. Козырев, А. Н. Социальный кредит в Китае: Обзор / А. Н. Козырев, И. В. Неволин // Цифровая экономика. – 2019. – № 3(7). – С. 70-74.
4. Турдаков Д. Ю. и др. Доверенный искусственный интеллект: вызовы и перспективные решения // Доклады Российской академии наук. Математика, информатика, процессы управления. – 2022. – Т. 508. – № 0. – С. 13–18.
5. Райков А.Н. Субъектность объяснимого искусственного интеллекта // Философские науки. 2022. Т. 65. № 1. С. 72–90.

Коваленко Михаил Андреевич, старший преподаватель, РТУ МИРЭА

kovalenko_m@mirea.ru

ORCID: 0000-0002-2233-6723

Ключевые слова

Доверие, эмоциональная окраска

Kovalenko M.A., Isaeva A.V., Emotional analysis of interaction between participants in the information system

Keywords

Trust, emotional coloring

DOI: 10.34706/DE-2024-01-11

JEL classification C61 – Методы оптимизации, программные модели, динамический анализ

Abstract

The article examines the problems of evaluating textual information in information systems with an emphasis on trust. The authors emphasize the importance of building trust according to various criteria, such as the credibility and expertise of the authors, the reliability of sources, clarity of presentation and compliance of information with needs. The article presents the formalization of the emergence of trust in the form of a python code.