# Вступ

Класифікація текстових документів є ключовим процесом у сучасному світі через великий обсяг текстової інформації. Метою цього реферату є огляд основних методів класифікації тексту, їх переваг та недоліків, а також вивчення практичних застосувань.

# Основні методи класифікації

## Методи на основі правил

Прості у реалізації, але обмежені в гнучкості. Приклад: Фільтрація спаму.

## Методи на основі статистичних моделей

- \*\*Наївний байєсівський класифікатор:\*\* швидкий, але припускає незалежність ознак.  
- \*\*Логістична регресія:\*\* враховує залежності між ознаками.

## Методи на основі машинного навчання

- \*\*SVM:\*\* ефективний для високовимірних даних.  
- \*\*KNN:\*\* простий, але повільний на великих наборах даних.  
- \*\*Рандомний ліс:\*\* висока точність і стійкість до переобучення.

## Методи на основі глибокого навчання

- \*\*RNN:\*\* добре обробляє послідовності даних.  
- \*\*Трансформери:\*\* висока точність у завданнях класифікації.

# Порівняння методів

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Метод | Простота реалізації | Швидкість виконання | Точність | Масштабованість | Вимоги до ресурсів |
| Правила | Висока | Висока | Низька | Низька | Низькі |
| Наївний Байєс | Висока | Висока | Середня | Висока | Низькі |
| Логістична регресія | Висока | Висока | Висока | Висока | Низькі |
| SVM | Середня | Середня | Висока | Висока | Середні |
| KNN | Висока | Низька | Висока | Низька | Середні |
| Рандомний ліс | Середня | Середня | Висока | Висока | Середні |
| RNN | Низька | Низька | Висока | Висока | Високі |
| Трансформери | Низька | Середня | Дуже висока | Висока | Дуже високі |

# Інструменти та бібліотеки

## Scikit-learn

Приклад коду на наївному байєсівському класифікаторі:

```python  
from sklearn.feature\_extraction.text import CountVectorizer  
from sklearn.naive\_bayes import MultinomialNB  
from sklearn.pipeline import Pipeline  
  
texts = ["Спам повідомлення з рекламою", "Нормальний лист від друга", "Ще одне спам повідомлення"]  
labels = ["spam", "ham", "spam"]  
  
model = Pipeline([  
 ('vectorizer', CountVectorizer()),  
 ('classifier', MultinomialNB())  
])  
  
model.fit(texts, labels)  
prediction = model.predict(["Це новий спам повідомлення"])  
print(prediction) # Вивід: ['spam']  
```

# Практичні приклади застосування

## Маркетинг

Аналіз відгуків клієнтів, автоматичне розподілення відгуків за категоріями.

## Медицина

Класифікація медичних записів для полегшення діагностики.

## Юриспруденція

Організація юридичних документів для прискорення роботи адвокатів та суддів.

# Висновок

Реферат розглянув методи класифікації текстових документів, їх переваги та недоліки. Ефективність залежить від конкретних завдань, обсягів та якості даних. Сучасні тенденції вказують на розвиток глибокого навчання та інтеграцію обробки природної мови в різні галузі.