УДК 004.8

АНАЛІЗ ВПЛИВУ ПОПЕРЕДНЬОЇ ОБРОБКИ ТЕКСТУ НА РЕЗУЛЬТАТИ ТЕКСТОВОЇ КЛАСИФІКАЦІЇ

Гущин І.В., Сич Д.О.
Харківський національний університет імені В.Н. Каразіна

Оглянута сутність основних підходів попередньої обробки тексту. Досліджено вплив попередньої обробки тексту на результати класифікації. На базі набору документів було проведено класифікацію як з використанням різноманітних підходів попередньої обробки тексту, так і без їх залучення. Були досліджені метрики точності класифікатора та час, затрачений на обробку документів. На основі результатів зроблені висновки актуальності використання попередньої обробки тексту у рамках класифікації.
Ключові слова: попередня обробка тексту, класифікація тексту, точність класифікації, машинне навчання, обробка природної мови.

Постановка проблеми. Сьогодення характеризується величезною кількістю інформації, об'єм якої невпинно зростає і давно вже не може бути охоплений без допомоги механізмів, що надають змогу на основі тих чи інших характеристик даних робити певні висновки та видавати спеціалістам або звичайним користувачам системи готовий результат обробки інформації. Це стосується статистичних даних, баз знань, систем пошуку тощо. Обробка тексту є одним із найнеобхідніших завдань, що вирішується за допомогою машинного навчання. Зокрема, доволі частою задачею є класифікація документів.

Зазвичай, текстові документи, що проходять скрізь механізми обробки інформації не є спеціально підготовленими. Це може впливати на час, що затрачується на опрацювання інформації, а також на фінальний результат.

Аналіз останніх досліджень і публікацій. Попередня обробка тексту використовується під час класифікації, кластеризації, автоматичному анотуванні документів тощо. Існує багато підходів – виділення стоп-слів, приведення регістру, стемінг, лематизація, розділення на n-грами тощо [2; 5]. У статі розглянуті та застосовані основні підходи, що широко використовуються у системах аналізу та обробки текстової інформації.

Виділення не вирішених раніше частин загальної проблеми. Безліч підходів з опрацювання тексту успішно використовуються в тих чи інших задачах і їх застосування з кожним днем все поширюється. Для використання цих інструментів необхідні ресурси – обчислювальні, часові тощо. Чи можна скоротити їх використання за допомогою попередньої обробки? Окрім того, постає питання – наскільки попередня обробка тексту покращує результати класифікації. У статті розглянуто основні підходи попередньої обробки тексту, проведені досліди щодо актуальності їх використання та на їх основі зроблені супутні висновки.

Мета статті. Головною метою цієї роботи є дослідити, наскільки попередня обробка тексту покращує результати роботи класифікатору та покращує загальну швидкість обробки текстових даних. Зробити висновок, чи дійсно необхідно використовувати ті чи інші підходи попередньої обробки та освітити результати їх застосування.

Виклад основного матеріалу. Обробка природної мови (natural language processing) – це комп'ютеризований підхід до аналізу тексту, що базується на низці теорій та наборі інструментів. Галузь включає в себе вирішення багатьох задач, наприклад: синтез мовлення, машинний переклад, інформаційний пошук тощо [1]. Зокрема, доволі частою задачею є класифікація текстових документів.

Класифікація текстових документів, так само як і в випадку класифікації об'єктів, полягає у віднесенні документа до одного з заздалегідь відомих класів. Часто класифікацію стосовно до текстових документів називають категоризацією або рубрикацією.

Для роботи класифікатора необхідна наявність готового набору класифікаційної інформації (наприклад, категорій з документами, векторів слів тощо) – датасет. Зазвичай, документи, що приймають участь у побудові датасету, а також документи, що є цільовими для класифікації, повинні пройти попередню обробку тексту. Попередня обробка тексту передбачає собою прибирання несуттєвої, зайвої інформації та уніфікації токенів.

Здебільшого, корпус досліджуваних документів включає в себе велику кількість слів. Деякі слова можуть не містити смислового навантаження, інші – можуть бути різноманітними варіантами одного слова тощо. Видалення надлишкових слів, а також приведення схожих між собою слів до однієї форми скорочують затрати часу на аналіз інформації. Усунення описаних проблем виконується на етапі попередньої обробки тексту [3].

Класичними рішеннями для видалення неінформативних слів та уніфікації слів зі схожим значенням є:

– видалення стоп-слів. Стоп-словами називаються слова, що є допоміжними в тексті та несуть мало інформації про зміст текстового документа. Зазвичай, списки таких слів складаються перед обробкою. В процесі обробки слова, що співпадають зі словами з цього списку, видаляються. Типовим прикладом таких слів є допоміжні слова та артиклі, наприклад: «так як», «окрім того» и тому подібні. Іноді, стоп-слова можуть бути специфічними для тієї чи іншої галузі та бути підібрані для конкретних випадків;

– стемінг – це морфологічний пошук основи слова. Сутність стемінгу полягає в перетворенні кожного слова до його нормальної форми. Нормальна форма виключає відмінок слова, множинну форму, особливості усного мовлення і т. п. Наприклад, слова "стиснення" і "стислий" повинні бути перетворені в корінну форму слова "стиск". Алгоритми мор-

фологічного розбору враховують мовні особливості і внаслідок цього є залежними від мови;

– лематизація – перетворення слова до його словникової форми. Цей механізм схожий зі стемінгом, але на відміну від нього, лематизація знаходить похідну форму слова, а не корінну. Тобто, слова "зеленого", "зелені", "зелененький" будуть приведені до слова "зелений";

– n-грами – це альтернатива морфологічному розбору і видаленню стоп-слів. N-грама – це частина рядка, що складається з N символів. Наприклад, слово "дата" може бути представлено 3-грамою "_да", "дат", "ата", "та_" або 4-грамою "_дат", "дата", "ата_", де символ підкреслювання замінює попередній або замикаючий слово пробіл. Порівняно зі стемінгом або видаленням стоп-слів, N-грами менш чутливі до граматичних і типографських помилок. Крім того, N-грами не вимагають лінгвістичної основи слів, що робить даний прийом більш незалежним від мови. Однак N-грами не вирішують проблему зменшення кількості неінформативних слів;

– приведення регістра. Цей прийом полягає в перетворенні всіх символів до верхнього або нижнього регістру. Наприклад, все слова "текст","Текст", "ТЕКСТ" приводяться до нижнього регістру – "текст" [2].

Для того, щоб зробити висновок наскільки добре працює класифікатор беруть до уваги його точність (accuracy) та міра f1 (f1 score).

Зазвичай, точність виражається у відношенні кількості правильно класифікованих документів до їх загальної кількості. Але лише на цю метрику покластися не можна, бо у разі незбалансованого датасету (документів одного класу набагато більше ніж документів іншого) значення може бути доволі високим, але насправді не буде відображати реального стану.

Міра f1 базується на значеннях влучності (precision) та повноти (recall). Precision відображає відношення кількості правильно класифікованих документів, що були віднесені до певного класу, до всіх документів, що були віднесені до цього класу. Recall – це доля знайдених класифікатором документів, що відносяться до певного класу, відносно всіх документів цього класу у датасеті [9].

Для того, щоб зрозуміти як впливає попередня обробка тексту проведемо дослід, у якому порівняємо час виконання підготовки тексту (токенізації, тобто розбиттю на окремі атомарні частини (токени) та векторизації – приведення тексту до числового вектору ознак) разом із техніками попередньої обробки та без них, час витрачений на навчання моделі, а також значення точності на f1 міри класифікатору.

Для виконання досліду було імплементовано програмне рішення [8] з використанням бібліотек sklearn та nltk [7; 8]. У якості датасету був обраний спеціальний набір новинних записів, що має назву 20 Newsgroups. 20 Newsgroups – це збірка приблизно 20 000 документів груп новин, розділених (майже) рівномірно на 20 різних класів. Колекція є популярним набором даних для експериментів у текстових програмах методів машинного навчання, таких як класифікація тексту та кластеризація тексту [4].

У досліді використовується класифікатор MultinomialNB (класифікатор Naive Bayes для багатовимірних моделей) та векторайзер CountVectorizer з декількома наборами вхідних параметрів, що впливають на обробку тексту. Для дослідження були змодельовані наступні ситуації:

– попередня обробка відсутня;

– приведення токенів до нижнього регістру;

– приведення токенів до нижнього регістру та видалення стоп слів;

– приведення токенів до нижнього регістру, видалення стоп слів та застосування механізму стемінгу та лемматизації (використовується SnowballStemmer та WordNetLemmatizer).

В результаті досліду повинні бути отримані результати метрик (accuracy, f1 score) та час, що був затрачений на навчання моделі та виконання векторизації. Дослід проводився на машині з наступними технічними показниками: процесор Intel Core i5, 6 GB оперативної пам'яті.

Висновки і пропозиції. Після проведення досліду із описаними вище інструментами були отримані результати, (таблиця 1) які відображають вплив попередньої обробки тексту на результати класифікації. Дані, що описують час виконання, були взяті як середній результат з десяти дослідів.

Час, що витрачається на векторизацію наданих документів не дуже відрізняється в перших трьох дослідах, однак, можна помітити, що наявна тенденція до мінорного зниження часу обробки. З цього можна зробити висновок, що під час приведення регістру кількість різноманітних токенів знижується, що надає змогу токенізувати меншу кількість слів. Видалення стоп-слів додатково зменшує кількість токенів, що дає також невелике прискорення в обробці. Час навчання моделі майже не відрізняється. Стандартні підходи. крім прискорення векторизації, покращують результати метрик оцінювання корисної дії класифікатора. На обраній виборці дані підходи дали зріст точності та міри f1, у середньому, близько 5%.

При додатковому використанні стемінгу та лематизації ми бачимо, що і результати метрик, і результати швидкості виконання не підвищи-

Таблиця 1

Дослід	Час виконання векторизації (сек.)	Час виконання навчання (сек.)	Міра f1	Точність
Без попередньої обробки	9.852	0.351	0.740	0.767
Приведення до нижнього регістру	8.550	0.301	0.745	0.773
Нижній регістр, видалення стоп-слів	8.820	0.304	0.779	0.802
Нижній регістр, видалення стоп-слів, стемінг	185.082	0.339	0.729	0.764
Нижній регістр, видалення стоп-слів, лемматизація	142.061	0.355	0.736	0.759

лись. Зокрема, час на векторизацію перевищує результати інших дослідів в десятки разів. Затрачений час пояснюється тим, що крім примітивної обробки слова виконується певний алгоритм, що приводить слово до основної форми. У разі великої вибірки це подовжує час обробки тексту. Також ми бачимо, що результати метрик не мають позитивної динаміки. Це може бути зумовлено особливостями обраного стемеру, мови документу тощо. Стемінг та лематизація, з одного боку, уніфікують словник, а з іншого – можуть приводити різні слова до однієї форми (наприклад, «зміст» та «змістили» будуть приведені до слова «зміст»). Чи означає це, що ці підходи не повинні використовуватися у попередній обробці? Зовсім ні. Але щоб вони приносили користь, ці механізми повиннні бути правильно підібрані (існує ряд інших стемерів, лематизаторів) та зконфігурований. У окремих випадках є релевантним написання власного стемінг алгоритму або використання специфічної бази знань для лематизатора. Також, щоб утримати зв'язок між словами документу, можна використати n-грами. Якщо час для виконання є критичним або ресурси машини не дозволяють користуватися згаданими механізмами, то можна сказати, що достатній рівень точності класифікації буде досягатися з використанням підходів приведення регістру та видалення стоп-слів.

Отже, можна зробити висновок, що, в цілому, вплив попередньої обробки тексту на результати класифікації є позитивним та має використовуватися під час роботи з текстом.

Список літератури:

1. Liddy E.D. 2001. Natural Language Processing. In Encyclopedia of Library and Information Science, 2nd Ed. NY. Marcel Decker, Inc. – 15 p.
2. Анализ данных и процессов: учеб. пособие / А.А. Барсегян, М.С. Куприянов, И.И. Холод, М.Д. Тесс, С.И. Елизаров. – 3-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2009. – 512 с.
3. Автоматическая обработка текстов на естественном языке и компьютерная лингвистика : учеб. пособие / Большакова Е.И., Клышинский Э.С., Ландэ Д.В., Носков А.А., Пескова О.В., Ягунова Е.В. – М.: МИЭМ, 2011. – 272 с.
4. 20 Newsgroups [Електронний ресурс]. – Режим доступу до ресурсу: http://qwone.com/~jason/20Newsgroups/. https://bitbucket.org/sych_d/text-preprocessing-investigation/src/master/.
5. Natural Language Processing with Python / Steven Bird, Ewan Klein, and Edward Loper. – O'Reilly Media, 2009. – 504 p.
6. Scikit-learn Machine Learning in Python [Електронний ресурс]. – Режим доступу до ресурсу: http://scikit-learn.org/stable/.
7. Natural Language Toolkit [Електронний ресурс]. – Режим доступу до ресурсу: https://www.nltk.org/.
8. Text preprocessing investigation code [Електронний ресурс]. – Режим доступу до ресурсу: https://bitbucket.org/sych_d/text-preprocessing-investigation/src/master/.
9. Text Data Management and Analysis / ChengXiang Zhai and Sean Massung. – ACM Books series, 2016. – 509 p.

Гущин И.В., Сыч Д.А.
Харьковский национальный университет имени В.Н. Каразина

АНАЛИЗ ВЛИЯНИЯ ПРЕДВАРИТЕЛЬНОЙ ОБРАБОТКИ ТЕКСТА НА РЕЗУЛЬТАТЫ ТЕКСТОВОЙ КЛАССИФИКАЦИИ

Аннотация
Рассмотрены основные подходы предварительной обработки текста. Исследовано влияние предварительной обработки текста на результаты классификации. На базе набора документов была проведена классификация как с использованием различных подходов предварительной обработки текста, так и без их привлечения. Были исследованы метрики точности классификатора и время, затраченное на обработку документов. На основе результатов сделаны выводы актуальности использования предварительной обработки текста в рамках классификации.
Ключевые слова: предварительная обработка текста, классификация текста, точность классификации, машинное обучение, обработка естественного языка.

Gushin I.V., Sych D.O.
V.N. Karazin Kharkiv National University

ANALYSIS OF THE IMPACT OF TEXT PREPROCCESSING ON THE RESULTS OF TEXT CLASSIFICATION

Summary
The main approaches of text preprocessing were investigated. The effect of text preprocessing on the classification results was investigated. On the basis of a range of documents, classification was performed with and without usage of text preprocessing. The accuracy metrics of the classifier and the time spent on processing documents were reviewed. Based on the results, conclusions were drawn on the relevance of the use of text processing in terms of text classification.
Keywords: text preprocessing, text classification, classification accuracy, machine learning, natural language processing.