How does the implementation of Natural Language Processing, particularly through keyword extraction algorithms like TF-IDF, enhance the efficiency and effectiveness of customer service in the digital era?

Contents

Introduction

Before the use of Natural Language Processing (NLP), the main workload for customer service staff was increasingly dealing with customer enquiries and problems and responding personally to customer needs in the form of primarily phone calls, and later email or text messages. Employees in direct contact with customers handled these inquiries manually. This approach was effective historically, but the volume of customer enquiries rose significantly with digitalisation, and the shift of the customer service function to online channels (Rust & Ming-Hui Huang, 2014). The escalating expectation for customer service's performance has stretched the function to its limits.

To uphold customer loyalty and contentment, it is imperative to address the issues arising from the increase in customer enquiries and to harness the development of technology to create value for the customer. The implementation of NLP in customer service is a strategic move to increase process efficiency and allow for greater utilisation of human engagement in value creation with AI.

Definition Natural Language Processing

NLP refers to the automated technology used to analyse and understand human language using computer algorithms (Kang et al., 2020). This has allowed researchers and companies to extract relevant insights from qualitative textual datasets more efficiently (Collobert et al., 2011).

NLP thereby combines the fields of computer science, linguistics, and mathematics with the goal of translating natural language into commands that can be understood by computer programs (Kang et al., 2020). NLP can be divided into two main tasks: Natural Language Understanding (NLU), which aims to extract information from written documents, and Natural Language Generation (NLG), which goal it is to generate natural language that can be understood by humans (McDonald, 1986; Schank, 1972).

Usually, NLP can be divided into three different steps. The first step is Text processing, which aims to increase the NLPs accuracy by cleaning the dataset from irrelevant symbols, spelling mistakes, or stop words (Kang et al., 2020). Next, the dataset must be represented. This means, that the natural language, which is unstructured data, gets translated into structured data in the form of numbers. This step enables different algorithms to read the data in later steps and perform tasks such as information extraction or text classification. Once the data is represented, it can be used to train a model or algorithm to solve a specific problem (Kang et al., 2020).

Function & application in the function (Chatbots in customer service)

NLP in customer service aims to improve customer experience and sales performance without significantly increasing the need for human labour (Kang et al., 2020). One way of doing so is the implementation of chatbots. Chatbots are conversational computer systems designed to mimic and engage in human conversation with users of a website to offer automated guidance and support (Caldarini et al., 2022). Chatbots rely on NLP, more specifically NLU and NLG, by taking natural language as an input and generating natural language as an output (Caldarini et al., 2022). Chatbots are capable of assisting multiple users simultaneously, making them both more efficient and inexpensive than a large workforce of customer support employees (Caldarini et al., 2022).

NLP-based chatbots are being used in customer service within a variety of industries and is ideal for automating repetitive, low-priority tasks that deplete employees' invaluable time, impeding their ability to deal with more intricate queries (Olujimi & Ade-Ibijola, 2023). Moreover, response capacity can be expanded to tackle a higher volume of requests and provide prompt and accurate solutions also outside the businesses' operating hours (Mariciuc, 2023). NLP can analyse and comprehend language, generating customised responses that align with the demand for personalisation which increases customer engagement (Schuetzler et al., 2020).

The focus of this report is on the NLU process, in particular keyword extraction. Keyword extraction falls under the field of text processing and is considered the initial phase of NLP. The objective is a comprehensive understanding of text content, structure, and meaning, to enable further natural language processing. This process entails analysing the occurrence and frequency of words in texts, documents, or queries, both within a single document and across multiple documents (Li, 2021). One commonly used algorithm to extract keywords is the TF-IDF method. In the following paragraph, the data cleaning and feature engineering process for keyword extraction will be analysed.

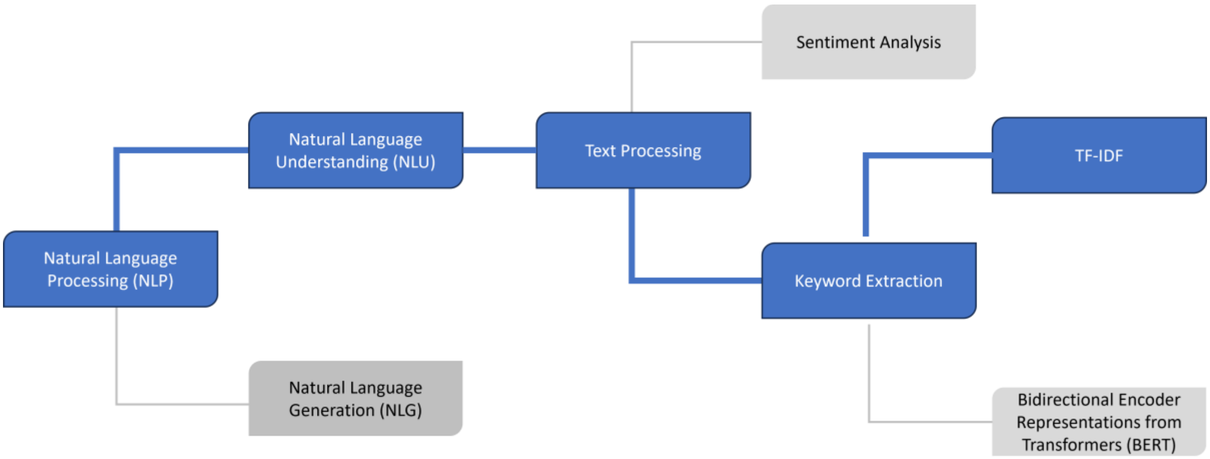

Figure 1: Report Scope

Data Cleaning and Feature Engineering

It can be argued that a problem of keyword extraction and NLP in general is the quality of the data provided. The effectiveness of keyword extraction is heavily dependent on the quality of the input data. Poorly structured, noisy, or unclean data can significantly hinder the process of accurately extracting keywords. Therefore, it is important to clean the data and pre-process it with feature engineering methods before using it for the algorithms (Lin & Hu, 2019).

Quality input data is critical for keyword extraction. Noise in form of irrelevant words, punctuation errors, or formatting issues can skew the results (Hsu & Chang, 2021). Cleaning the data by removing such noise enhances the accuracy and relevance of the extracted keywords. Moreover, the data often comes from diverse sources and exhibits inconsistencies, such as different spellings, abbreviations, or terminological variations. Standardising the text through data cleaning is crucial to maintain consistency, enabling algorithms to identify and evaluate the importance of keywords, thus making the method more efficient and accurate.

Texts like customer requests, while not necessarily faulty, should still be cleaned prior to being used for keyword extraction. Particularly in customer service contexts, texts often include irrelevant standard phrases, greetings, or other essentially useless information. Cleaning data to remove such content allows for a greater focus on the text's significant parts, which are more likely to yield valuable keywords This also reduces computational complexity by removing redundant information (Qian et al., 2021).
With data cleaning being an important step, feature engineering is used to transform the raw text data into a structured format that is more suitable for analysis by algorithms. This process significantly enhances keyword extraction's effectiveness and NLPs in various aspects. Firstly, there are supporting features that enable algorithms to better understand the data, like part-of-speech tagging, where each word in a sentence is tagged with the correct part of the speech in a grammatical sense, indicating the behaviour of the words (Martinez, 2012). Furthermore, a commonly used method to gather information is Tokenization. Text is split into smaller units, which could be paragraphs or smaller, with the smallest unit being a letter. This breaking of text enables more distinct words. These units are referred to as tokens (Bhirud et al., 2019).
Furthermore, it is noteworthy that customer service input is often received in the form of numerous data types. Firstly, chatbots primarily receive nominal data in textual form through customer requests or categorical, such as user choices in pre-defined menus. Secondly among further types, interval and ratio data, which are usually collected when users provide account numbers, dates, and other quantified information. These different types require different types of preparation to be efficiently processed by the algorithms. (Coenders & O'Loughlin, 2004). For example, nominal data typically needs to be processed into a numerical type through methods like encoding to be understood and processed by the algorithms (Casari & Zheng, 2018). Having explored the diverse types of data encountered in customer service and necessary preprocessing steps for each to facilitate efficient algorithmic processing, the following section will now scrutinize the application of the Term Frequency-Inverse Document Frequency (TF-IDF).

Presentation of the chosen model

The TF-IDF concept is an algorithm for keyword extraction and information retrieval that is also used in customer service chatbots (Wang & Ning, 2020). The algorithm is based on a weighted statistical method and applied to a set or a corpus of documents. For each document, TF determines the most frequently used words or phrases and weights their importance based on the number of times they appear. IDF is then applied to evaluate the frequency of the same words or phrases in the corpus. If they are equally common in the other documents, the algorithm determines low importance. These words include articles, prepositions and common fillers (Thampi et al., 2021). If a phrase or word with a high frequency in a single document cannot be found with the same frequency in the rest of the corpus, the algorithm assumes that the word or phrase is distinctive and characteristic of the original document and that the word or phrase is representative of the text. On this basis, it can then categorise the text and assign appropriate responses to the query (Barton & Peuker, 2022).

Applying the algorithm to a customer service situation, there might be a customer query about a damaged product. It could be "My product is damaged. The interior and exterior are damaged". The query appears in a corpus of 100 texts in which 13 other queries mention the word "damaged".

To calculate the TF-value of a word or phrase, the following formula can be applied:

$$TF_{ij} = \frac{n_{ij}}{\sum_k n_{kj}}$$

Here, n_{ij} is the number of occurrences of a certain word or phrase t_i in the query d_j. It is divided by the sum of the frequency of occurrences of all words or phrases n_{kj} in the text d_j.

$$TF = \frac{2}{10}$$

In the example, the TF for the word "damaged" is 0.2.

IDF can then be calculated as follows:

$$IDF_i = \log\left(\frac{|D|}{|\{j : t_i \in d_j\}|}\right)$$

Here, $|D|$ is the total number of documents in the corpus. $|\{j : t_i \in d_j\}|$ stands for the total amount of texts in which word or phrase t_i appears under the condition that $n_{ij} \neq 0$. In the calculation, the logarithm of the quotient is obtained to simplify the amount of multiplication calculations.

$$IDF = \log\left(\frac{100}{14}\right)$$

In the example, the IDF for the word "damaged" is approximately 0.8547.

Hence, the algorithm can be calculated as:

$$TF - IDF_{ij} = TF_{ij} \times IDF_i$$

Therefore, a high frequency of a word or phrase in a text combined with a low frequency of the same word or phrase in the whole corpus of texts will result in a high weight score. The algorithm is therefore able to identify words or phrases that are highly representative of the text and filter out less important words that occur with high frequency across the set of documents (Yao et al., 2019; Firoozeh et al., 2020; Li, 2021; Thampi et al., 2021).

In the example, the TF-IDF for the word "damaged" is approximately 0.17094.

In code, the TF-IDF may be expressed as follows:

```
In [8]: import pandas as pd
        import math
        from collections import Counter

        def compute_tf(text):
            word_count = Counter(text)
            total_words = len(text)
            tf = {word: count / total_words for word, count in word_count.items()}
            return tf

        def compute_idf(documents):
            document_count = len(documents)
            word_presence = {word: sum(1 for doc in documents if word in doc) for word in set(word for doc in documents for word in doc)
            idf = {word: math.log(document_count / (1 + presence)) for word, presence in word_presence.items()}
            return idf

        def compute_tfidf(tf, idf):
            tfidf = {word: tf_value * idf.get(word, 0) for word, tf_value in tf.items()}
            return tfidf

        documents = ["Can you help me track my order I haven't received any shipping information",
                     "What is your return policy I'd like to know before making a purchase",
                     "Do you offer international shipping What are the shipping costs and delivery times",
                     "I forgot my password How can I reset it ",
                     "Are there any ongoing promotions or discounts available ",
                     "Can you provide more details about the product specifications ",
                     "Is this product currently in stock ",
                     "What payment methods do you accept",
                     "How do I cancel my order It was a mistake, and I need assistance ",
                     "I received a damaged item What should I do for a replacement or refund ",
                     "Can you help me choose the right size Do you have a size chart ",
                     "What is the status of my refund I returned an item a few days ago",
                     "Are there any additional costs like taxes or duties on international orders",
                     "Do you have a customer loyalty program or rewards program",
                     "Can I change the shipping address for my order",
                     "What is your customer support phone number I need to speak with someone directly",
                     "Is it possible to add an item to my order after it has been placed",
                     "Can you recommend similar products based on my recent purchase",
                     "How do I unsubscribe from your newsletter or promotional emails",
                     "What security measures do you have in place to protect my payment information"
                     ]

        tokenized_documents = [doc.lower().split() for doc in documents]

        tf_matrices = [compute_tf(doc) for doc in tokenized_documents]

        idf = compute_idf(tokenized_documents)

        tfidf_matrices = [compute_tfidf(tf, idf) for tf in tf_matrices]

        df_tfidf = pd.DataFrame(tfidf_matrices)

        df_tfidf
```

Figure 2: TF-IDF Code

Out[8]:

my	order	i	haven't	received	...	recent	unsubscribe	from	newsletter	promotional	emails	security	measures	place	protect
.061424	0.106638	0.061424	0.177122	0.145932	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
.088723	NaN	0.177446	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
.057036	0.099021	0.114073	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	0.114073	NaN	0.135509	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
.053234	NaN	0.053234	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
.088723	0.154033	0.088723	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	0.057036	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
.053234	0.092420	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
.079851	NaN	NaN	NaN	NaN	...	0.230259	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	0.079851	NaN	NaN	...	NaN	0.230259	0.230259	0.230259	0.230259	0.230259	NaN	NaN	NaN	NaN
.061424	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	0.177122	0.177122	0.177122	0.177122

Figure 3: TF-IDF Code Output

This code defines functions for calculating TF, IDF and TF-IDF. Next, a sample of customer queries randomly generated is tokenised to separate the terms. This allows words to be analysed individually (Badlani et al., 2021). The algorithm is then applied to the dataset and the results are displayed in a Pandas data frame. Examples of relatively high scores are highlighted in green.

In chatbot engineering, TF-IDF is used to categorise queries. The system uses TF-IDF and other techniques such as cosine similarity to find the most appropriate response to a user query from a knowledge base provided to it (Badlani et al., 2021).

To exemplify the practical application of the TF-IDF algorithm in the customer service function, the following comparison illustrates 100 randomly generated customer queries using a word cloud in Python (the generation codes can be found in the Appendix). The first illustration displays the queries separated by word without further cleaning. Notable words include "Can" or "I". The TfidfVectorizer feature extraction method from the sklearn.feature_extraction.text library was employed in the second image to apply the TF-IDF algorithm, which converts the queries into TF-IDF matrix features. The difference between the words displayed in the second image and those of the simple word cloud is evident. Words such as "order" and "policy" are now prevalent. This illustrates that keyword extraction using the TF-IDF algorithm has a significant impact on the rest of the natural language processing process and provides an important basis for generating a satisfactory response to the customer.

Figure 24: Unprocessed Wordcloud

Figure 5:3 Wordcloud with applied TF-IDF algorithm

Tools/Software Recommendation

Since it is challenging to confirm which specific companies are using keyword extraction relying on the TF-IDF algorithm, specific examples cannot be provided due to proprietary technologies and privacy policies (Suhaili et al., 2021).
Thus, while many companies have implemented chatbots with underlying algorithms, such as keyword extraction (TF-IDF), some still rely on manual customer service. For example, when interacting with the customer service of the online marketplace "Prozis", any requests customers have, are processed by human workforce. Requests, such as tracking of an order, therefore take approximately 24 hours to be answered (Prozis, 2023).

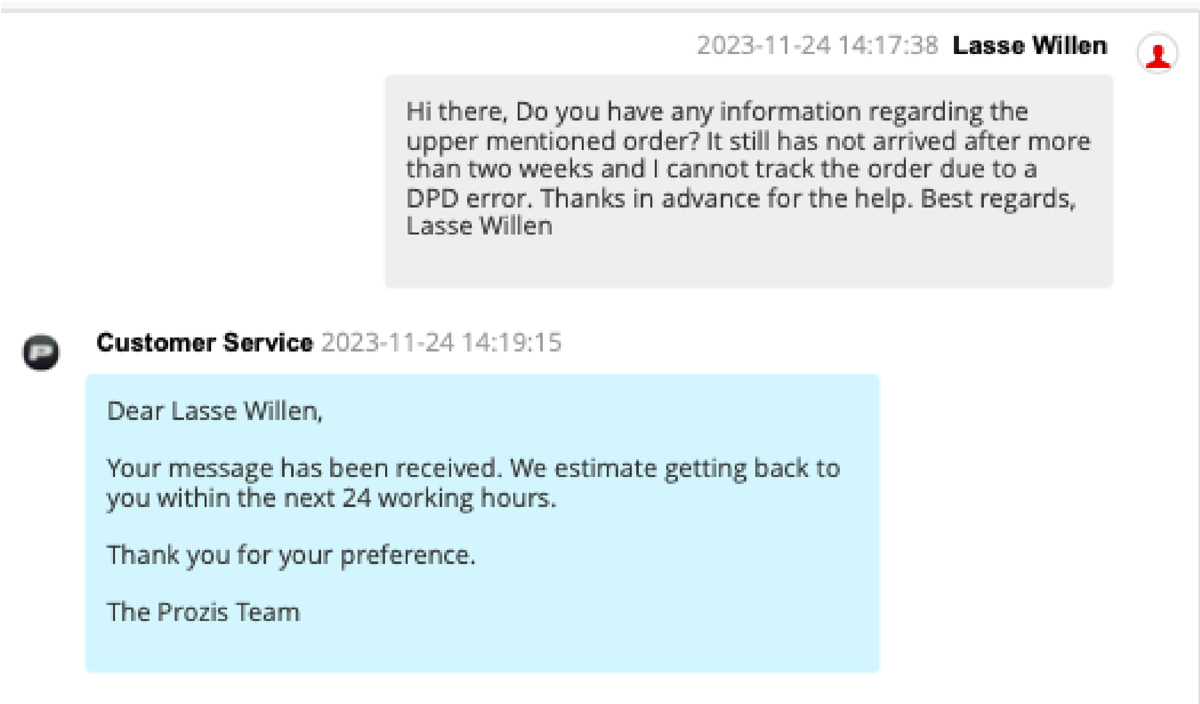

Figure 6: Current Customer Service of "Prozis" Webshop

This process could be significantly improved through the implementation of a chatbot that analyses customer requests and provides answers with the underlying data from the company's internal database. This would likely improve customer satisfaction through reduced waiting times and also save valuable resources for the company. If the chatbot is not equipped to deal with the customer's request, a human workforce could then still solve the problem but would not be needed for simpler requests. These disadvantages of not relying on data-driven methods, such as keyword extraction, hold true for customer services in various companies.
Due to various research done in the field of NLP, there are many different papers that prove that keyword extraction, including the TF-IDF algorithm, is successfully used in chatbots.

Identification of Limitations

Although the TF-IDF algorithm is highly effective in classifying different words based on their frequency, the algorithm is not capable of classifying the words based on their meaning. In addition, TF-IDF is incapable of considering the location of certain words within a text. Furthermore, TF-IDF does not recognize the importance of certain words (Zhang & Ge, 2019). The algorithm is further limited to the early-stage process of text processing within NLP and does not present a holistic method. Further processing is required in order to build a complete chatbot workflow and answer a customer query. Analysing it from a broader perspective, while NLP-based chatbots have changed the landscape of customer service severely in recent years, there are some limitations that challenge the effectiveness of using methods like keyword extraction. The human language is multifaceted and nuanced; thus polysemy, homonyms and synonyms or irony can cause data ambiguities for the algorithms used which lead to misunderstanding of customer queries and consequently diminished customer experience (Olujimi & Ade-Ibijola, 2023). Evaluating different keywords is also language-specific and can influence the algorithm. English for example is a low inflectional language, thus performance is higher for keyword extraction. Roman languages, however, have a high inflection and therefore require specialised approaches (Firoozeh et al., 2020). Furthermore, contextuality poses a challenge for NLP-based chatbots as customer queries often consist of multiple responses and interactions. Context might be lost, or previous customer input is not integrated into the interpretations and response to a query (Aslam, 2023). Firoozeh et al. (2020) further argue that the importance of specific target elements in the textual units can vary for target user and application processes and it is often unclear which targeted units are “key” elements for an extraction.

Conclusion

To conclude, the integration of NLP, particularly through the application of keyword extraction algorithms such as TF-IDF, represents a significant advancement in the realm of customer service. This transformation, driven by the increasing digitalisation and online engagement, has fundamentally altered the way customer queries and interactions are managed. The ability of NLP-based chatbots to process and analyse large volumes of customer data in real-time has revolutionised the efficiency and effectiveness of customer service operations.

The implementation of algorithms like TF-IDF in customer service chatbots has demonstrated substantial benefits. It enables the extraction of relevant information from customer interactions, paving the way for more personalized and responsive service. This not only enhances customer satisfaction but also streamlines the workload on customer service teams, allowing them to focus on more complex tasks that require human intervention.

Looking ahead, the potential of NLP in customer service is vast. The ongoing developments in AI and machine learning promise even more sophisticated chatbot capabilities, further improving the accuracy and contextuality of responses. However, challenges such as language nuances, cultural variations, and the need for continuous model training remain. Addressing these challenges will be crucial for the next generation of NLP applications in customer service.

Sources

Aslam, F. (2023). The Impact of Artificial Intelligence on Chatbot Technology: A Study on the Current Advancements and Leading Innovations. *European Journal of Technology*, *7*(3), Article 3. https://doi.org/10.47672/ejt.1561

Badlani, S., Aditya, T., Dave, M., & Chaudhari, S. (2021). Multilingual Healthcare Chatbot Using Machine Learning. *2021 2nd International Conference for Emerging Technology (INCET)*, 1–6. https://doi.org/10.1109/INCET51464.2021.9456304

Barton, T., & Peuker, A. (2022). Extraktion und Analyse von Schlüsselwörtern für eine automatisierte Literaturauswertung zum Thema Empfehlungssysteme. *HMD Praxis der Wirtschaftsinformatik*. https://doi.org/10.1365/s40702-022-00909-1

Bhirud, N., Tataale, S., Randive, S., & Nahar, S. (2019). A literature review on chatbots in healthcare domain. International journal of scientific & technology research, 8(7), 225-231.

Caldarini, G., Jaf, S., & McGarry, K. (2022). A Literature Survey of Recent Advances in Chatbots. *Information*, *13*(1), 41. https://doi.org/10.3390/info13010041

Casari, A., & Zheng, A. (2018). Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists. *O'Reilly Media*.

Coenders, G., & O'Loughlin, C. (2004). Estimation of the European Customer Satisfaction Index: Maximum Likelihood versus Partial Least Squares. Application to Postal Services. *Total Quality Management & Business Excellence* https://doi.org/10.1080/1478336042000255604

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural Language Processing (Almost) from Scratch. *Journal of Machine Learning Research*, *12*, 2493–2537.

Firoozeh, N., Nazarenko, A., Alizon, F., & Daille, B. (2020). Keyword extraction: Issues and methods. *Natural Language Engineering*, *26*(3), Article 3. https://doi.org/10.1017/S1351324919000457

Hsu, I.-C., & Chang, C.-C. (2021). Integrating machine learning and open data into social Chatbot for filtering information rumor. *Journal of Ambient Intelligence and Humanized Computing, 12, 1023–1037*. https://doi.org/10.1007/s12652-020-02119-3

Juglan, K. C., Sharma, B., Gehlot, A., Singh, S. P., Hussein, A., & Alazzam, M. B. (2023). Exploring the Effectiveness of Natural Language Processing in Customer Service. *2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE),* 814–818. https://doi.org/10.1109/ICACITE57410.2023.10183107

Kang, Y., Cai, Z., Tan, C.-W., Huang, Q., & Liu, H. (2020). Natural language processing (NLP) in management research: A literature review. *Journal of Management Analytics*, *7*(2), 139–172. https://doi.org/10.1080/23270012.2020.1756939

Li, J. (2021). A comparative study of keyword extraction algorithms for English texts. *Journal of Intelligent Systems*, *30*(1), 808–815. https://doi.org/10.1515/jisys-2021-0040

Lin, J.-R., & Hu, Z.-Z. (2019). Understanding On-Site Inspection of Construction Projects Based on Keyword Extraction and Topic Modeling. *2019 6th International Conference on Systems and Informatics (ICSAI)* https://doi.org/10.1109/ACCESS.2020.3035214

Martinez, A. R. (2012). Part-of-speech tagging. *WIREs Comp Stat*, *4,* 107–113. https://doi.org/10.1002/wics.195

McDonald, D. D. (1986). *Natural Language Generation*. https://apps.dtic.mil/sti/pdfs/ADA175135.pdf

Mariciuc, D. F. (2023). A Bibliometric Analysis of Publications on Customer Service Chatbots. *Management Dynamics in the Knowledge Economy*, *11*(1), Article 1. https://doi.org/10.2478/mdke-2023-0004

Olujimi, P. A., & Ade-Ibijola, A. (2023). NLP techniques for automating responses to customer queries: A systematic review. *Discover Artificial Intelligence*, *3*(1), Article 1. https://doi.org/10.1007/s44163-023-00065-5

Prozis. (2023). *Customer Service*. https://www.prozis.com/ie/en/customerservice/index

Qian, H., Li, X., Zhong, H., Guo, Y., Ma, Y., Zhu, Y., Liu, Z., Dou, Z., & Wen, J.-R. (2021). *Pchatbot: A Large-Scale Dataset for Personalized Chatbot*. https://doi.org/10.1145/3404835.3463239

Rust, R. T. & Ming-Hui Huang. (2014). The Service Revolution and the Transformation of Marketing Science. *Marketing Science*, *33*(2), Article 2. https://doi.org/10.1287/mksc.2013.0836

Schank, R. C. (1972). Conceptual dependency: A theory of natural language understanding. *Cognitive Psychology*, *3*(4), 552–631. https://doi.org/10.1016/0010-0285(72)90022-9

Schuetzler, R. M., Grimes, G. M., & Scott Giboney, J. (2020). The impact of chatbot conversational skill on engagement and perceived humanness. *Journal of Management Information Systems*, *37*(3), Article 3. https://doi.org/10.1080/07421222.2020.1790204

Suhaili, S. M., Salim, N., & Jambli, M. N. (2021). Service chatbots: A systematic review. *Expert Systems With Applications*, *184*, 957–4174. https://doi.org/10.1016/j.eswa.2021.115461

Thampi, S. M., Gelenbe, E., Atiquzzaman, M., Chaudhary, V., & Li, K.-C. (Eds.). (2021). *Advances in Computing and Network Communications: Proceedings of CoCoNet 2020, Volume 2* (Vol. 736). Springer Singapore. https://doi.org/10.1007/978-981-33-6987-0

Tiwari, R., Saxena, M. K., Mehendiratta, P., Vatsa, K., Srivastava, S., & Gera, R. (2018). Market segmentation using supervised and unsupervised learning techniques for E-commerce applications. *Journal of Intelligent and Fuzzy Systems*, *35*(5), 5353–5363. https://doi.org/10.3233/JIFS-169818

Wang, X., & Ning, H. (2020). TF-IDF Keyword Extraction Method Combining Context and Semantic Classification. *Proceedings of the 3rd International Conference on Data Science and Information Technology*, 123–128. https://doi.org/10.1145/3414274.3414492

Yao, L., Pengzhou, Z., & Chi, Z. (2019). Research on News Keyword Extraction Technology Based on TF-IDF and TextRank. *2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS)*, 452–455. https://doi.org/10.1109/ICIS46139.2019.8940293

Zhang, T., & Ge, S. S. (2019). An Improved TF-IDF Algorithm Based on Class Discriminative Strength for Text Categorization on Desensitized Data. *Proceedings of the 2019 3rd International Conference on Innovation in Artificial Intelligence*, 39–44. https://doi.org/10.1145/3319921.3319924

Appendix

Code for Wordcloud without algorithm:

```
import matplotlib.pyplot as plt
import nltk
nltk.download('punkt')
from wordcloud import WordCloud
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
import nltk
nltk.download('stopwords')
```

```
# List of queries
queries = [
    "How can I track my order?",
    "What's your return policy?",
    "Do you offer international shipping?",
    "Can I change my shipping address?",
    "What payment methods do you accept?",
    "Are there any ongoing promotions or discounts?",
    "Can you help me reset my password?",
    "Is this product in stock?",
    "What are the shipping costs and delivery times?",
    "Can I cancel my order?",
    "I received a damaged item. What should I do?",
    "Do you have a size chart?",
    "Can you provide more details about product specifications?",
    "What is the status of my refund?",
    "Are there any additional costs for international orders?",
    "Can I add items to my order after it's been placed?",
    "How do I contact customer support?",
    "What's your customer support phone number?",
    "Is there a warranty on your products?",
    "Can you recommend similar products to what I've purchased?",
    "What security measures do you have for my payment information?",
    "Do you have a loyalty program?",
    "How do I unsubscribe from your newsletter?",
    "Are gift cards available?",
    "Can you provide more information about your company's history?",
    "What is your price matching policy?",
    "Are there any restrictions on using discount codes?",
    "Can I return an item without the original packaging?",
    "Is there a restocking fee for returns?",
    "What's the difference between standard and expedited shipping?",
    "Can I order by phone?",
    "How do I check the status of my order?",
    "What is your chatbot's name?",
    "Can you help me find a specific product on your website?",
    "Do you offer a student discount?",
    "Is it possible to change the color or size of my order before it ships?",
    "Can I get a refund if the price drops after I've made a purchase?",
    "What information do you need for a warranty claim?",
    "Do you have a mobile app?",
    "How do I sign up for your rewards program?",
    "Are there any restrictions on using gift cards?",
    "What is your email address for customer inquiries?",
    "Can I order online and pick up in-store?",
    "How do I leave a product review on your website?",
```

```
    "Do you have a live chat option for customer support?",
    "What is your policy on price adjustments for sale items?",
    "Can you provide more information about your sizing options?",
    "Do you offer a satisfaction guarantee?",
    "What is your policy on out-of-stock items?",
    "How do I redeem a promo code?",
    "Can you tell me more about your eco-friendly practices?",
    "Are there any restrictions on using multiple promo codes on one order?",
    "What is your policy on pre-orders?",
    "Can you help me with a product that's not listed on your website?",
    "Do you have a physical store location?",
    "What is your policy on price gouging?",
    "Can I request a price adjustment for a sale item I purchased recently?",
    "How do I sign up for product notifications?",
    "Do you have a size guide for clothing items?",
    "Can I change the delivery date for my order?",
    "What is your policy on back-ordered items?",
    "Can I order a gift card online?",
    "Do you have a mobile app for tracking orders?",
    "What is your policy on returns for personalized items?",
    "Can I exchange an item for a different color or size?",
    "How do I apply a store credit to my order?",
    "What is your policy on price matching with competitors?",
    "Can I purchase a warranty for my product after the initial purchase?",
    "Do you offer a bulk discount for large orders?",
    "Can I return an online purchase to a physical store location?",
    "What is your policy on holiday returns?",
    "Can I modify my order after it has been placed?",
    "Do you have a size guide for shoes?",
    "How do I check the balance on my gift card?",
    "Can I request a gift receipt with my order?",
    "What is your policy on lost or stolen gift cards?",
    "Can you provide more information about the materials used in your products?",
    "Do you offer a trade-in program for old products?",
    "How do I track my package if it shows as delivered but I haven't received it?",
    "Can you recommend a product for a specific use case?",
    "What is your policy on price adjustments for clearance items?",
    "Do you have a newsletter I can subscribe to for updates and promotions?",
    "How do I update my account information?",
    "Can I use a promo code on a gift card purchase?",
    "What is your policy on delayed shipments?",
    "Can I return an item without the original receipt?",
    "Do you offer a military discount?",
    "How do I initiate a warranty claim for a defective product?",
    "What is your policy on rain checks for out-of-stock items?",
    "Can I place a bulk order for a corporate event?",
    "How do I apply for a job at your company?",
    "Can you provide more information about the fit of your clothing?",
    "Do you offer a repair service for damaged products?",
    "What is your policy on canceled orders due to product unavailability?",
    "Can I request a custom order for a product not available on your website?",
    "How do I unsubscribe from text message notifications?",
    "Do you have a waitlist for out-of-stock items?",
    "Can I return an item purchased with a gift card for a cash refund?",
    "What is your policy on delayed deliveries due to weather conditions?",
    "Can I change the shipping method for my order after it's been placed?"
]
```

```
# Combine all customer enquiries into a single string
text = ' '.join(customer_enquiries)

# Tokenize the text
tokens = word_tokenize(text)


# Combine the filtered tokens into a string for the word cloud
filtered_text = ' '.join(tokens)

# Generate the word cloud
wordcloud = WordCloud(width=800, height=400, background_color='white', stopwords=set()).generate(filtered_text)

# Display the word cloud using matplotlib
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
```

Wordcloud without algorithm:

Code for Wordcloud with TF-IDF algorithm applied:

```python
from sklearn.feature_extraction.text import TfidfVectorizer
from wordcloud import WordCloud
import matplotlib.pyplot as plt
import numpy as np

# List of queries
queries = [
    "How can I track my order?",
    "What's your return policy?",
    "Do you offer international shipping?",
    "Can I change my shipping address?",
    "What payment methods do you accept?",
    "Are there any ongoing promotions or discounts?",
    "Can you help me reset my password?",
    "Is this product in stock?",
    "What are the shipping costs and delivery times?",
    "Can I cancel my order?",
    "I received a damaged item. What should I do?",
    "Do you have a size chart?",
    "Can you provide more details about product specifications?",
    "What is the status of my refund?",
    "Are there any additional costs for international orders?",
    "Can I add items to my order after it's been placed?",
    "How do I contact customer support?",
    "What's your customer support phone number?",
    "Is there a warranty on your products?",
    "Can you recommend similar products to what I've purchased?",
    "What security measures do you have for my payment information?",
    "Do you have a loyalty program?",
    "How do I unsubscribe from your newsletter?",
    "Are gift cards available?",
    "Can you provide more information about your company's history?",
    "What is your price matching policy?",
    "Are there any restrictions on using discount codes?",
    "Can I return an item without the original packaging?",
    "Is there a restocking fee for returns?",
    "What's the difference between standard and expedited shipping?",
    "Can I order by phone?",
    "How do I check the status of my order?",
    "What is your chatbot's name?",
    "Can you help me find a specific product on your website?",
    "Do you offer a student discount?",
    "Is it possible to change the color or size of my order before it ships?",
    "Can I get a refund if the price drops after I've made a purchase?",
    "What information do you need for a warranty claim?",
    "Do you have a mobile app?",
    "How do I sign up for your rewards program?",
    "Are there any restrictions on using gift cards?",
    "What is your email address for customer inquiries?",
    "Can I order online and pick up in-store?",
    "How do I leave a product review on your website?",
```

```
    "Do you have a live chat option for customer support?",
    "What is your policy on price adjustments for sale items?",
    "Can you provide more information about your sizing options?",
    "Do you offer a satisfaction guarantee?",
    "What is your policy on out-of-stock items?",
    "How do I redeem a promo code?",
    "Can you tell me more about your eco-friendly practices?",
    "Are there any restrictions on using multiple promo codes on one order?",
    "What is your policy on pre-orders?",
    "Can you help me with a product that's not listed on your website?",
    "Do you have a physical store location?",
    "What is your policy on price gouging?",
    "Can I request a price adjustment for a sale item I purchased recently?",
    "How do I sign up for product notifications?",
    "Do you have a size guide for clothing items?",
    "Can I change the delivery date for my order?",
    "What is your policy on back-ordered items?",
    "Can I order a gift card online?",
    "Do you have a mobile app for tracking orders?",
    "What is your policy on returns for personalized items?",
    "Can I exchange an item for a different color or size?",
    "How do I apply a store credit to my order?",
    "What is your policy on price matching with competitors?",
    "Can I purchase a warranty for my product after the initial purchase?",
    "Do you offer a bulk discount for large orders?",
    "Can I return an online purchase to a physical store location?",
    "What is your policy on holiday returns?",
    "Can I modify my order after it has been placed?",
    "Do you have a size guide for shoes?",
    "How do I check the balance on my gift card?",
    "Can I request a gift receipt with my order?",
    "What is your policy on lost or stolen gift cards?",
    "Can you provide more information about the materials used in your products?",
    "Do you offer a trade-in program for old products?",
    "How do I track my package if it shows as delivered but I haven't received it?",
    "Can you recommend a product for a specific use case?",
    "What is your policy on price adjustments for clearance items?",
    "Do you have a newsletter I can subscribe to for updates and promotions?",
    "How do I update my account information?",
    "Can I use a promo code on a gift card purchase?",
    "What is your policy on delayed shipments?",
    "Can I return an item without the original receipt?",
    "Do you offer a military discount?",
    "How do I initiate a warranty claim for a defective product?",
    "What is your policy on rain checks for out-of-stock items?",
    "Can I place a bulk order for a corporate event?",
    "How do I apply for a job at your company?",
    "Can you provide more information about the fit of your clothing?",
    "Do you offer a repair service for damaged products?",
    "What is your policy on canceled orders due to product unavailability?",
    "Can I request a custom order for a product not available on your website?",
    "How do I unsubscribe from text message notifications?",
    "Do you have a waitlist for out-of-stock items?",
    "Can I return an item purchased with a gift card for a cash refund?",
    "What is your policy on delayed deliveries due to weather conditions?",
    "Can I change the shipping method for my order after it's been placed?"
]

# Create a TfidfVectorizer
vectorizer = TfidfVectorizer(stop_words='english')

# Fit and transform the queries
tfidf_matrix = vectorizer.fit_transform(queries)

# Get feature names (terms)
feature_names = vectorizer.get_feature_names_out()

# Combine TF-IDF scores across all queries
total_tfidf_scores = np.sum(tfidf_matrix, axis=0)

# Create a dictionary with terms and their corresponding total TF-IDF scores
terms_tfidf = dict(zip(feature_names, total_tfidf_scores.A[0]))

# Create a WordCloud
wordcloud = WordCloud(width=800, height=400, background_color='white').generate_from_frequencies(terms_tfidf)

# Display the WordCloud using matplotlib
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
```

Wordcloud with TF-IDF algorithm applied: