International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V72I11P133

Volume 72 Issue 11, 345-356, November 2024

Original Article

Personalization of Query Model for Search Keywords in Context of Corporate Users Using Machine Learning Techniques

T. B. Lalitha[1], S. Gokila[2]

[1,2]*Department of Computer Application, Hindustan Institute of Technology and Science, Chennai, India.*

[1]*Corresponding Author : lalitha.srm@gmail.com*

Received: 24 August 2024 Revised: 11 November 2024 Accepted: 14 November 2024 Published: 29 November 2024

Abstract *- Effective information retrieval is crucial for strategic decision-making, market analysis, and competitive intelligence in the corporate sector. In the contemporary digital era, the efficiency and relevance of search queries significantly impact user satisfaction and productivity, especially within corporate environments. The proposed query model aims to enhance the precision and relevance of search results by systematically structuring keywords and utilizing advanced search techniques to enhance user experience for corporate users. This includes a comprehensive framework that integrates user-specific data, such as historical search patterns, interaction metrics, and contextual information, to tailor search results to individual user preferences and organizational needs. By employing state-of-the-art machine learning algorithms, including collaborative filtering, natural language processing, and supervised learning models, the proposed approach aims to improve the precision of search results and the relevance of recommendations. Through rigorous experimentation and evaluation of corporate datasets, demonstrate the effectiveness of the personalized query model in optimizing search performance and user satisfaction. The efficacy of this query model is demonstrated through its application to various corporate research scenarios, including market trends analysis, competitor benchmarking, and regulatory compliance monitoring. The results indicate a significant improvement in the accuracy and comprehensiveness of retrieved data, thereby supporting more informed decision-making processes in the corporate industry. This study contributes to the growing body of research on personalized search systems and the field of information science by providing a structured approach to keyword query formulation and implementing machine learning-based solutions tailored specifically for the needs of corporate professionals.*

Keywords *- Query model, KNN, SVM, Machine learning, User information data, Search keyword.*

1. Introduction

In the rapidly evolving landscape of the corporate industry, access to precise and relevant information is pivotal for informed decision-making, strategic planning, and maintaining a competitive edge. The proliferation of digital information and the complexity of data sources necessitate the development of advanced methods for information retrieval. A personalized query model for search keywords represents a strategic approach to address this need, offering tailored solutions that align with specific corporate objectives and contexts. Personalization of query models involves customizing the search process to accommodate different corporate entities' unique needs and characteristics. The main goal of search personalization [23] is to enhance the user experience by delivering customized results that better align with an individual's specific interests and requirements. This entails a nuanced understanding of industry-specific terminology, the identification of pertinent data sources, and applying sophisticated search techniques. The effectiveness of information retrieval can be substantially enhanced by refining keyword queries to reflect the particularities of a company's operational environment, market position, and strategic goals. The corporate industry encompasses various sectors with unique terminology, data needs, and information sources. For instance, a technology firm may prioritize innovation trends, patents, and market share queries, whereas a financial institution may focus on regulatory updates, economic indicators, and competitive analysis. Therefore, personalizing query models to align with the specific requirements of different corporate entities is crucial for optimizing information retrieval outcomes. This requirement delves into the methodologies and principles underlying the personalization of query models for search keywords in the corporate sector. It outlines the steps in crafting a personalized query model, including identifying core concepts relevant to the corporate context, expanding with industry-specific synonyms and related terms, and using user information data to refine search parameters. Additionally, it emphasizes the

importance of incorporating various forms and variations of keywords and specific criteria such as geographic location, time period, and regulatory considerations. In conclusion, personalizing query models for search keywords significantly advances corporate information retrieval. By aligning search strategies with the specific needs of corporate entities, it is possible to achieve greater precision and relevance in search results, thereby supporting more effective decision-making processes in the corporate industry. This paper aims to provide a comprehensive framework for developing and implementing such personalized query models, contributing to the evolution of information retrieval practices in the corporate sector.

2. Background and Related Works

In the digital age, corporations are inundated with vast amounts of data generated from many sources, including internal databases, market reports, regulatory updates, and social media platforms. The challenge lies in effectively sifting through this data to extract pertinent information to solve strategic problems in a minimum timeline. Conventional search methods frequently fail to provide accurate and relevant results, resulting in inefficiencies and lost opportunities. This is where the personalization of query models for search keywords becomes crucial. Search personalization is rooted in information retrieval, which seeks to improve how users find and access information. Personalization tailors the search process to the user's specific context, preferences, and needs. In the corporate industry, this means adapting search queries based on user profiles [22] to reflect the unique characteristics of different sectors, companies, and even individual roles within those companies to gain personal growth and enhance problem solutions. For example, personalized search models recognize that the search information needs of a financial analyst differ significantly from those of a marketing executive or a legal advisor.

The evolution of search technologies has paved the way for more sophisticated and nuanced query models. Early search engines relied heavily on keyword matching, often resulting in irrelevant results. Advances in Natural Language Processing (NLP), machine learning, and artificial intelligence have enabled the development of more complex algorithms that can understand context, semantics, and user intent. These technologies are fundamental to creating personalized search experiences with higher precision and relevance.

In the corporate sector, personalized query models must address several key challenges:

1. Diverse Information Needs [6] [7]: Corporations operate in various industries, each with its terminology, regulations, and market dynamics. A personalized query model must account for these differences to provide relevant results.
2. Dynamic Data Sources [8] [9]: The relevant information sources continually evolve, including new publications, market reports, social media trends, and regulatory updates. Personalized search models need to adapt to these changes swiftly.
3. User-Specific Contexts [10] [11]: Individual users within a corporation may have different roles, responsibilities, and information needs. For instance, a CEO might seek strategic insights, while a compliance officer requires detailed regulatory information. Personalized queries must cater to these varied contexts.
4. Complex Queries [12] [13]: Corporate searches often involve complex queries that combine multiple keywords, Boolean operators, and specific criteria such as time frames or geographic locations. Effective personalization involves managing this complexity to yield valuable results.

To address these challenges, a personalized query model for search keywords in the corporate industry typically involves several key components:

1. Core Concept Identification [14] [15]: Defining the main subjects or themes relevant to the corporate context.
2. Expansion with Synonyms and Related Terms [16] [17]: Incorporating various synonyms, related terms, and industry-specific jargon to capture a broad spectrum of relevant information.
3. Boolean Operators [18]: Using Boolean operators like AND, OR, and NOT to refine search logic and improve precision.
4. Keyword Variations [19]: Considering different forms and variations of keywords to ensure comprehensive coverage.
5. Specific Criteria [20]: Adding criteria such as geographic location, time period, and industry-specific parameters to tailor the search to specific needs.
6. Wildcards and Truncation [21]: Employing symbols to include various word forms and spellings, enhancing query flexibility.

By integrating these components, personalized query models can significantly enhance the relevance and accuracy of search results.

This, in turn, supports more effective decision-making processes within the corporate industry, enabling professionals to access the precise information they need promptly and efficiently. The personalization of query models represents a critical advancement in information retrieval. As corporations continue to navigate an increasingly data-driven world, efficiently and accurately accessing relevant information will be a key determinant of success. These studies are related to the personalization of query models for search keywords in various fields of work, identified through the limited knowledge acquired from the literature survey. Sijin P et al. [1] present a method called Conceptualization with Typed Terms of Query (CTTQ) aimed at improving the effectiveness of keyword searches. It proposes utilizing a

conceptual network to understand user query intentions better, enhancing search results' relevance and accuracy. The method incorporates randomized machine learning algorithms to optimize query suggestions and evaluates their effectiveness using the Normalized Discounted Cumulative Gain (nDCG), demonstrating improved usefulness of query suggestions. Additionally, the research highlights its relevance in the context of Industry 4.0, addressing the challenges posed by digital transformation and the need for AI-driven solutions. Overall, the study marks a significant advancement in information retrieval by focusing on user-centered search processes.

Lívia Kelebercová et al. [2] research analyses search queries related to COVID-19 to understand the public interest and the dissemination of information during the pandemic. The study employs keyword extraction techniques to identify significant trends and differences in search behavior, particularly distinguishing between true and false news. The findings reveal statistically significant differences in total mean TQ (Trend Quotient) values for various keywords associated with the pandemic, indicating that false news often garnered more attention than true news for certain topics. Manish Kumar et al. [3] discuss advancements in web crawling techniques aimed at improving the retrieval of relevant information for the query from the internet. The authors highlight search engines' challenges, such as the vast volume of online data and the need for efficient methods to filter and prioritize web pages based on user queries. They propose innovative approaches, including keyword-based query-focused crawlers that guide the crawling process through metadata and relevance feedback mechanisms.

The paper also emphasizes the importance of generating priority keywords from relevant web pages to enhance the effectiveness of the search crawling process. Overall, the document illustrates how technology is evolving to better meet users' information needs by refining the methods used in web crawling. X. Meng et al. [4] introduce a novel approach to spatial keyword queries that aims to enhance the efficiency and relevance of results in Location-Based Services (LBS). It addresses the limitations of existing models, which often focus solely on textual similarity and location proximity, neglecting semantic relationships and numerical attributes. To overcome these challenges, the work proposed using Conditional Generative Adversarial Nets (CGAN) to expand original query keywords into semantically related terms, allowing for more comprehensive results even when the original keywords are rare. Additionally, developed a new hybrid index structure called AIR-tree, which integrates location, text, and numerical information, facilitating efficient query processing that considers both semantic matching and numerical attributes through a Skyline-based approach. Experimental validation on real Point of Interest (POI) datasets demonstrates that the proposed method significantly improves execution efficiency and user satisfaction compared to existing methods, ultimately contributing to a more effective spatial keyword querying process in LBS. S. Kim et al. [5] discuss the development of a meta-suggestion engine for search queries. It explains the calculation of query similarity, the refinement process using three factors, and the determination of the optimal cut-off value for query suggestions. The meta-suggestion algorithm outlines the step-by-step process of how the candidate queries are refined and sorted to provide optimal query suggestions to the user. It involves applying three factors sequentially to enhance the quality of the suggestions and ensure the user receives relevant and accurate query recommendations.

It is presented along with the design and implementation details of the engine as a browser extension. The importance of query suggestions in enhancing search efficiency is highlighted, emphasizing the impact on user experience and search process acceleration. This paper provides a detailed framework for developing and implementing personalized query models, underscoring their importance in meeting the unique demands of the corporate sector.

3. Proposed Work and Methodology

The Personalization of query module aims to systematically organize and address the needs and demands of self-directed learners by creating a detailed and individualized learning experience based on their individual backgrounds and preferences. This process involves several key steps:

3.1. Learner Background Profiling

- Basic Profiles: Collecting essential information about the learner, such as name, age, gender, qualification, and occupation. This data provides a foundational understanding of the learner's demographic and professional context.
- Level of Involvement: Assessing the learner's engagement level could range from casual interest to deep commitment to the subject matter.
- Prior Knowledge and Learning Sessions: Evaluating the learner's existing knowledge and previous learning experiences to appropriately tailor the new learning journey.

3.2. Needs and Demands Analysis

- Specific Request Details: Gathering detailed information about what the learner needs explicitly. This includes understanding the exact nature of their inquiry or learning goal.
- Motivation and Purpose: Identifying why the learner is pursuing this particular learning path. This could involve understanding their personal, professional, or academic motivations.
- Implementation Context: Determining where and how the learner intends to apply the acquired knowledge. This helps in aligning the learning content with practical applications.

- Role Specifications: Considering the learner's current role or anticipated role within an organization or field may influence the relevance and depth of the content provided.

3.3. Estimation of Learning Abilities

- *Skill Assessment*: Conducting a precise estimation of the learner's abilities based on the collected background information and specific needs. This involves evaluating their learning pace, comprehension level, and preferred learning style.

3.4. Profile Creation

- Using the extracted data, a comprehensive learner profile is created. This profile serves as a personalized roadmap for the learner's educational journey, highlighting their strengths, areas for improvement, and tailored learning objectives.

3.5. Questionnaire Evaluation

- *Evaluation Process*: The learner undergoes a questionnaire-based evaluation tailored to their specific request. This assessment is designed to gauge their current knowledge level and understanding of the subject matter.
- *Grading Scale*: Based on the questionnaire results, the learner's knowledge level is evaluated from 0 to 10 points on a grading scale. This scale clearly indicates their proficiency and areas that require further attention.

By integrating these steps, the Personalization of query module aims to provide a highly customized and effective learning experience that aligns with the individual learner's needs, motivations, and capabilities. This approach enhances the relevance and impact of the learning process. It supports the learner in achieving their specific goals in a structured and efficient manner using a methodical approach to problem-solving. Figure 1 illustrates the flow of this Personalization of the query module. This module aims to identify search keywords or phrases from the user's profile features that closely match the user's current profile when searching for specific learning content on the website. This allows for generating a personalized query request based on the search keywords. This personalized query request is used when the user searches for specific learning content on the website, enhancing their search experience. This functional module primarily comprises two key components: the User Information Database and the K-Nearest Neighbour (KNN) algorithm. These components work together to personalize the user's search experience and provide relevant content based on their profile and search queries.

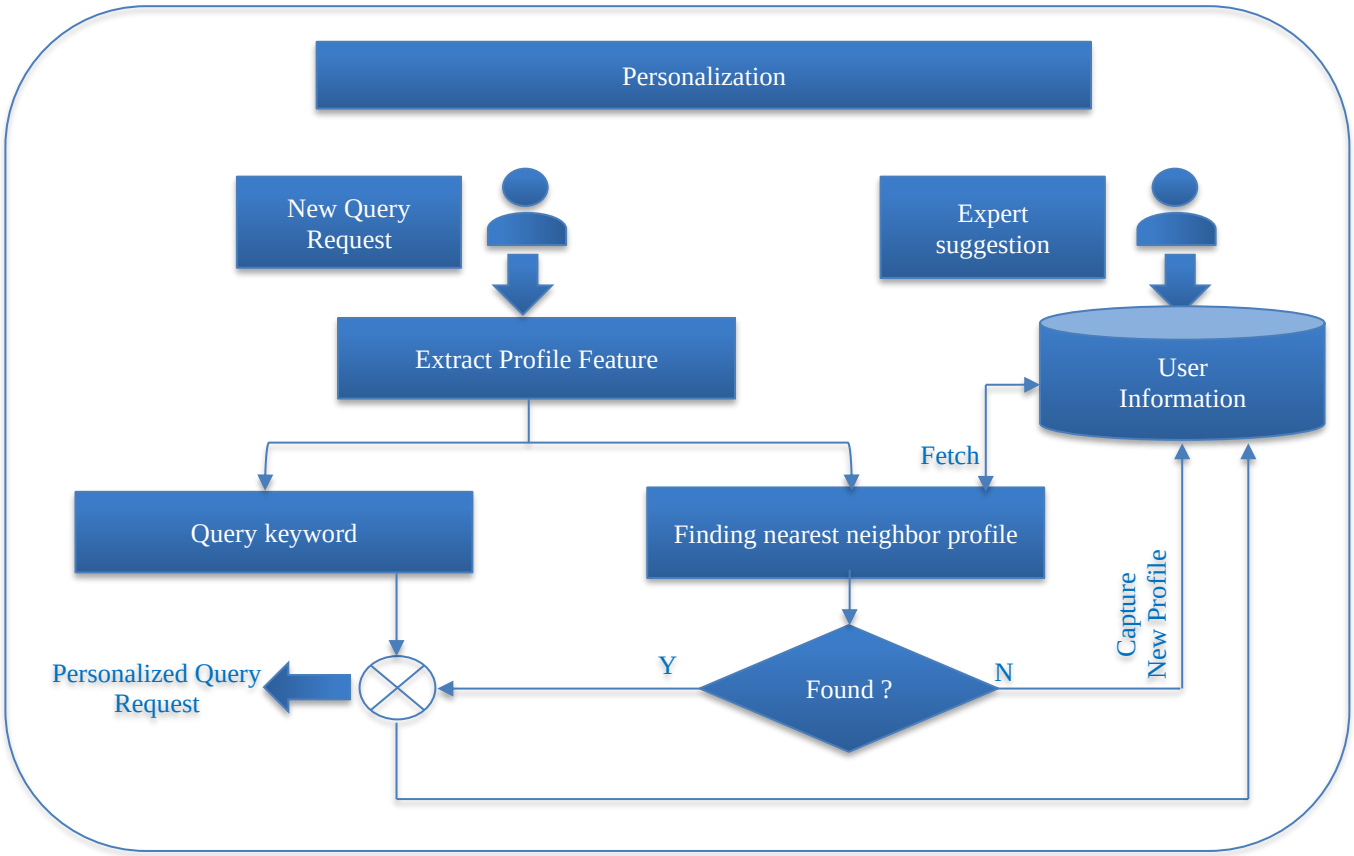

Fig. 1 Personalization of query model

3.6. User Information Database [25]

- Storage of Profiles: This database stores the profiles of previously visited learners, capturing detailed information such as their background, search history, and learning preferences.
- Incorporation of Expert Opinions: Expert insights related to specific subject knowledge are also included to provide a comprehensive view of the skill sets or prerequisites necessary for the learner's particular query.

Various algorithms, such as K-Nearest Neighbors (KNN) [26], Polynomial SVM, RBF SVM, and Sigmoid SVM, are implemented for comparative analysis to identify the best fit. The process involves:

- Profile Matching: The algorithm is used to identify the nearest neighbor profiles from the user information repository based on the learner's profile information.
- Knowledge-Based Filtering: This technique helps refine the search by filtering through the profiles and identifying those closely matching the learner's current needs and demands.

The current working process involves the below steps.

Profile Information Utilization: The learner's profile information is used to identify the nearest neighbor profiles from the user information repository using the KNN algorithm and knowledge-based filtering.

Database Construction: The user information repository stores profiles of previously visited learners and incorporates expert opinions. This repository offers a comprehensive view of the skill set or prerequisites necessary for the learner's particular query.

Profile Matching and Query Generation

- If a similar profile for the learner is found in the repository, the learner's profile is matched with these nearest neighbor profiles.
- If a similar profile is not found, the learner's profile is captured and stored in the database for future reference.
- A personalized query request is generated based on the learner's background profile, estimated performance level, and the skill set of their nearest neighbors' profiles.

Search Enhancement: The personalized query request is used when the learner searches for specific learning content on the website. This ensures that the search results are highly relevant to the learner's needs and preferences, enhancing their search experience. By leveraging the User Information Database, the K-Nearest Neighbour algorithm, and SVM algorithms, this module aims to provide a highly personalized search experience, ensuring that learners find the most relevant content based on their individual profiles and search queries.

3.6.1. User Information form

The User Information Database is established by gathering user data through standard questionnaires during registration. This dataset includes user profile details like profession, qualifications, job title, years of experience, and knowledge level required, among other factors. The user information form is illustrated in Figure 2.

Data Collection

Users fill out a questionnaire that collects comprehensive information during the registration process. This includes:

- Profession: The user's current job role or field of work.
- Qualifications: Educational background and any relevant certifications.
- Job Title: The specific title the user holds in their current job.
- Years of Experience: The duration of the user's professional experience.
- Knowledge Level: Self-assessed proficiency in various relevant areas or subjects.

Security and Privacy [32]

To ensure the security and privacy of the User Information Database, robust measures must be implemented to prevent any unauthorized access or data leaks to external sources. This includes:

- Encryption: All data stored in the database should be encrypted to protect it from unauthorized access.
- Access Control: Strict access control policies should be in place to ensure that only authorized personnel can access sensitive data.
- Routine Audits: Conduct periodic security audits to uncover and address potential vulnerabilities.

Data Archival Policy [33]

The platform must have a well-defined data archival policy to ensure compliance with legal and regulatory requirements. This includes:

- Retention Periods: Defining how long different types of data should be retained.
- Data Deletion: Procedures for securely deleting data that is no longer needed.
- Compliance: Ensuring data handling practices comply with relevant data protection regulations (e.g., GDPR, CCPA).

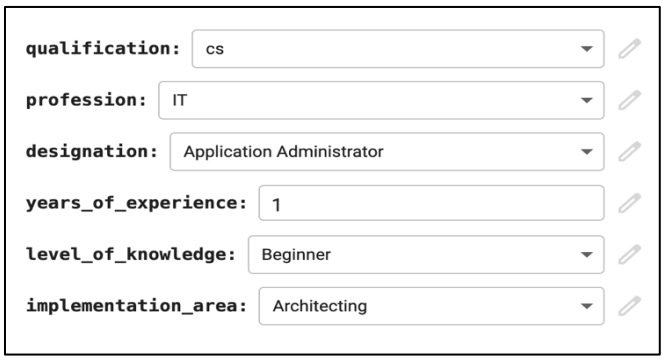

Fig. 2 User information form

Database Structure and Content

Figure 3 serves as a sample reference to visualize the structure and content of the database. It provides an overview of how user information is organized within the database, including various profile features and details collected from registered users. This reference aids in understanding the database's composition and role in the overall system.

By organizing user information in this structured manner, the platform can efficiently manage and leverage user data to personalize the learning experience. The combination of robust security measures and a clear data archival policy ensures that user data is protected and managed in compliance with legal requirements.

This is a multi-class classification problem where the data points are evenly distributed across the classes. The target feature, 'level_of_content,' is defined with 'xlabel' as 'level_of_content' and 'ylabel' as 'count.' Figure 4 gives a clear scenario of content levels 0, 1, and 2, which can be described as low, medium and high.

	profession	designation	years_of_experience	qualification	level_of_knowledge	level_of_content	implementation_area	keywords
0	IT	Program Manager	20	cs	Expert	High	management	program strategy
1	IT	Program Manager	21	cs	Expert	High	management	program charter
2	IT	Program Manager	22	cs	Expert	High	management	benefit management
3	IT	Program Manager	19	cs	Intermediate	Medium	management	benefit management
4	IT	Program Manager	19	cs	Expert	High	management	risk management
...	...	...	...	...	...	...	...	...
453	IT	PMO	2	cs	Beginner	Low	Operations	leave monitoring
454	IT	PMO	4	cs	Intermediate	Medium	Operations	compliance monitoring
455	IT	PMO	2	cs	Beginner	Low	Operations	compliance monitoring
456	IT	PMO	5	cs	Expert	High	Operations	project communications
457	IT	PMO	3	cs	Intermediate	Medium	Operations	project communications

458 rows × 8 columns

Fig. 3 User information database

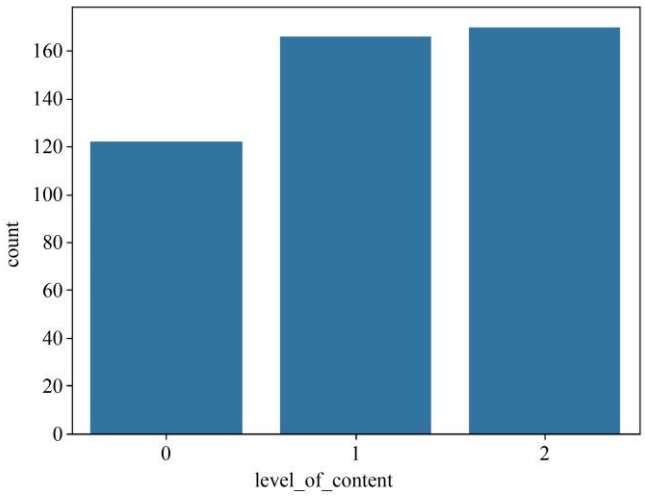

Fig. 4 Content levels

3.7. Applying K- Nearest Neighbor's Algorithm

The K-Nearest Neighbors (KNN) algorithm [26] is a supervised machine learning technique extensively utilized for regression and classification tasks. The core idea of KNN is to classify an object based on the majority vote of its neighbors. Essentially, an object is assigned to the most common class among its k-nearest neighbors, where "k" is a user-defined parameter representing the number of neighbors to consider.

This makes KNN a simple yet powerful algorithm frequently employed in recommendation systems, pattern recognition, and anomaly detection applications. The detailed explanation of concepts involved in the K-Nearest Neighbors (KNN) Algorithm is as follows.

Concept

- Classification: In classification tasks, KNN assigns a class label to an object based on the majority class of its k-nearest neighbors.
- Regression: In regression tasks, KNN predicts the value of an object based on the average or weighted average of the values of its k-nearest neighbors.

Parameter "k"

The value of "k" is crucial and can affect the algorithm's performance. A smaller "k" value can lead to noisy predictions, while a larger "k" value can smooth out the predictions but might ignore local patterns.

Distance Metric [34]

The algorithm relies on a distance metric (such as Euclidean distance) to find the nearest neighbors. The selection of the distance metric can also influence the performance. The detailed explanation of Implementation with KNeighborsClassifier for the query model is as follows.

Model Training

The `KNeighborsClassifier` model uses the feature' level of content' as the target variable. This involves fitting the model to the training data, where each data point has a known class label.

Evaluation Metrics [34]

- Accuracy: Evaluates the proportion of correctly classified instances relative to the total number of instances.
- Precision: Indicates the proportion of true positive results in the predicted positive instances.
- Recall: It refers to sensitivity and measures the proportion of true positive results among all actual positive instances.
- F1 Score: The harmonic mean of precision and recall, offering a single metric that balances both aspects.

Confusion Matrix

A confusion matrix is created to evaluate the performance of the classification model. This N × N matrix (where N is the number of target classes) provides a detailed breakdown of:

- True Positives (TP): Positive instances that are correctly predicted.
- True Negatives (TN): Negative instances that are correctly predicted.
- False Positives (FP): Incorrectly predicted positive instances (Type I error).
- False Negatives (FN): Incorrectly predicted negative instances (Type II error).

Performance Metrics Calculation [35]

The values in the confusion matrix are used to calculate key performance metrics:

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)}$$

$$\text{Precision} = \frac{TP}{\text{TP} + \text{FP}}$$

$$\text{Recall} = \frac{TP}{\text{TP} + \text{FN}}$$

$$\text{F1 score} = 2 * \frac{Precision*Recall}{\text{Precision+Recall}}$$

High values for accuracy, precision, recall, and F1 score indicate that the model performs well, effectively classifying instances and minimizing errors.

In the context of personalizing search queries for learners on a website, the KNN algorithm can be utilized as follows:

- Profile Matching [27, 28]: User profiles are compared to find the k-nearest neighbors based on features like profession, qualifications, and learning preferences.
- Personalized Query Generation [29, 30]: The model predicts the most relevant content or learning materials for the user based on the profiles of similar users (nearest neighbors).
- Performance Monitoring [31]: The performance of the KNN model is continuously monitored using the aforementioned metrics, ensuring that the recommendations remain accurate and relevant.

By implementing KNN with a focus on robust evaluation and continuous improvement, the personalized query module can significantly enhance the search experience for users, providing them with content that closely matches their needs and preferences.

3.8. Polynomial SVM Algorithm [36]

Polynomial SVM (Support Vector Machine) is a variant of the SVM algorithm that uses a polynomial kernel function to transform the input data into a higher-dimensional space.

This allows the algorithm to handle non-linear relationships between the features, making it suitable for complex classification tasks where the decision boundary is not straight. The degree of the polynomial kernel determines the flexibility of the decision boundary, with higher degrees allowing for more complex boundaries.

3.9. RBF SVM Algorithm [37] [38]

RBF SVM (Radial Basis Function Support Vector Machine) is an SVM algorithm that uses the Radial Basis Function as its kernel. The RBF kernel maps the input data into a higher-dimensional space, enabling the algorithm to handle non-linear relationships between features effectively. This kernel is particularly powerful for classification tasks where the decision boundary is complex and non-linear. The RBF SVM is known for its flexibility, as it can adapt to various data distributions, making it a popular choice for many machine learning tasks. The key hyperparameters in RBF SVM are the regularization parameter (C) and the kernel coefficient (gamma), which together control the trade-off between maximizing the margin and minimizing classification errors.

3.10. Sigmoid SVM Algorithm [24, 39]

Sigmoid SVM (Support Vector Machine with a Sigmoid Kernel) is a variant of the SVM algorithm that uses the sigmoid function as its kernel. The sigmoid kernel is often seen as similar to a neural network with a two-layer perceptron, where the decision boundary is formed by a combination of the input features passed through the sigmoid activation function.

This kernel allows the SVM to model non-linear relationships between the features, making it useful in scenarios where a linear boundary cannot separate the data. However, the sigmoid kernel is less commonly used than the RBF or polynomial kernels, as it can sometimes struggle with convergence and perform poorly on complex datasets. The sigmoid kernel is controlled by two key parameters: the scale of the input features and the offset, which determine the shape and flexibility of the decision boundary.

4. Results and Discussion

4.1. KNN Algorithm

K-Nearest Neighbors (KNN) is a classification algorithm that identifies and leverages the most similar data points in a training dataset to classify new, unseen instances. The method relies on a distance metric to gauge the similarity between data points, and it classifies a query instance based on the class labels of its ‘K’ nearest neighbors. The parameter ‘K’ is crucial as it determines the number of neighbors considered in the classification decision, and the user defines it before applying the algorithm. The first step in applying KNN involves pre-processing the data. Specifically, categorical variables—such as profession, designation, qualification, level of knowledge, content, and implementation area—must be converted into numerical values. This conversion is necessary because distance metrics, which form the basis of the KNN algorithm, require numerical inputs to accurately compute the similarity between data points. Once the data is appropriately formatted, it is divided into two subsets: the training and test sets. This separation facilitates the assessment of the model’s performance. The training set is used to build

the KNN model, while the test set is reserved for assessing its effectiveness. Within this framework, an input feature matrix, denoted as 'X,' is established, encompassing all the features used for classification tasks. A target column vector 'y' is also defined, representing the target labels or classes associated with each data point. The KNeighborsClassifier model is constructed and applied to the training data with this setup. For a given value of 'K' (e.g., K = 1), the classifier predicts the class of new instances based on the majority vote of the nearest neighbors. The mean error is calculated by comparing the predicted values with the actual values from the test set to evaluate the model's accuracy and robustness. This process involves iterating over various values of 'K,' often ranging up to a specified limit, such as K = 40, to determine the optimal value that minimizes the prediction error and enhances the classifier's performance.

Overall, KNN is a straightforward yet powerful algorithm that relies on the similarity between data points to make predictions. By pre-processing categorical data, dividing the dataset, and iteratively evaluating the model across different 'K' values, KNN provides a flexible and effective approach to classification tasks. Figure 5 presents a plot illustrating the relationship between mean error and various K values in the K-Nearest Neighbors (KNN) algorithm. The optimal K value is identified as the one that yields the lowest mean error, which is critical for ensuring accurate classification outcomes. This optimal K value balances model complexity and performance, enhancing the classifier's effectiveness. After training the KNeighborsClassifier model, specifically configured with 'level of content' as the target feature, the model is applied to predict the outcomes for new test samples. These test samples represent new user profiles not part of the training dataset. If the predicted values (denoted as (y_{pred}}) align with the true values (denoted as (y_{true}), it indicates that the features of the new user profile closely match those in the existing User Information Database. Such congruence suggests that the system has correctly identified and classified the new user profile.

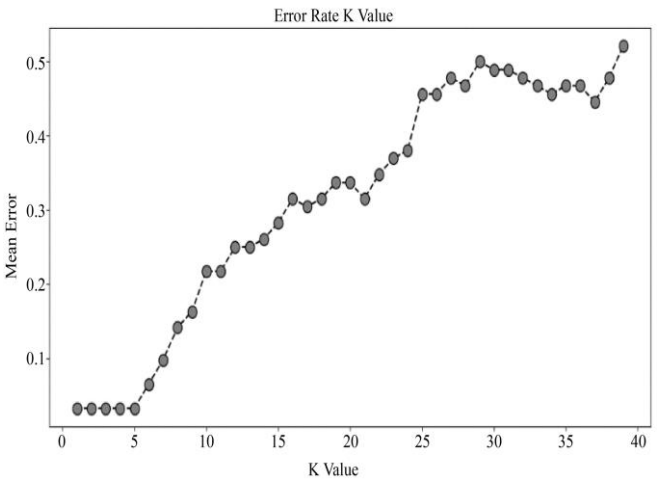

Fig. 5 Plot of mean error versus K value

Table 1. Metrics of evaluation

Metric	Value
Accuracy	0.87
F1 Score	0.88
Precision	0.85
Recall	0.91

Following the model's training, its performance is assessed using various metrics, including Accuracy, F1 Score, Precision, and Recall. These metrics provide a comprehensive evaluation of the model's classification capabilities. The results for these metrics are summarized in Table 1, which provides a detailed account of the model's effectiveness in predicting the 'level of content' for the user profiles.

4.2. Polynomial SVM Algorithm

To evaluate the performance of the Polynomial SVM model, a heatmap of the confusion matrix (`cm_poly`) is generated using the `sns.heatmap` function, with annotations enabled to display the values within each cell. This visual representation in Figure 6 helps us understand the model's classification results.

Based on the previous context, the heatmap you provided visualizes the confusion matrix for a classification model, likely the RBF SVM. In the heatmap of the confusion matrix, the x-axis represents the predicted classes, while the y-axis denotes the actual classes.

The diagonal cells extending from the top left to the bottom right of the matrix illustrate the number of correctly classified instances for each class. These diagonal entries indicate the model's correct predictions. Conversely, the off-diagonal cells reflect the misclassified instances, highlighting the discrepancies between the predicted and actual classes.

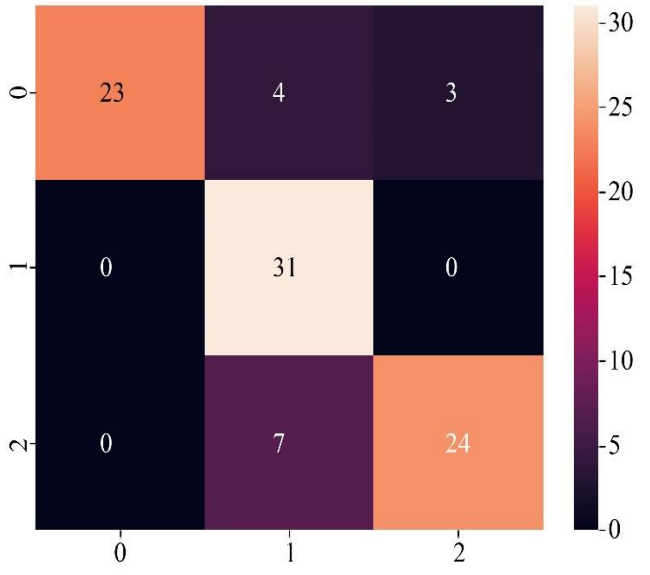

Fig. 6 Confusion matrix for Polynomial svm

	precision	recall	f1-score	support
0	1.00	0.77	0.87	30
1	0.74	1.00	0.85	31
2	0.89	0.77	0.83	31
accuracy			0.85	92
macro avg	0.88	0.85	0.85	92
weighted avg	0.87	0.85	0.85	92

Fig. 7 Evaluation of polynomial svm model

This visual representation helps assess the model's performance and identify areas where errors may be made. The breakdown of classes is classified as follows:

- Class 0: Out of 30 actual instances, 23 were correctly predicted as class 0, while 4 were incorrectly classified as class 1 and 3 as class 2.
- Class 1: All 31 instances were correctly classified as class 1, with no misclassifications.
- Class 2: Out of 31 instances, 24 were correctly predicted as class 2, and 7 were incorrectly classified as class 1.

This heatmap suggests that the model performs well, particularly with class 1, but there are some misclassifications, especially between classes 0 and 2. Next, the `accuracy_score` function is used to calculate the accuracy of the Polynomial SVM model by comparing the predicted labels (`poly_pred`) against the actual test labels (`y_test`). The accuracy score is then printed as "0.8478260869565217" to measure how well the model performed overall quickly. Finally, the `classification_report` function generates a detailed report of the model's performance, including metrics such as precision, recall, and F1-score for each class. This report in Figure 7 is printed to offer a comprehensive view of how the model performed across all classes.

4.3. RBF SVM Algorithm

To assess the performance of the RBF SVM model, a heatmap of the confusion matrix (`cm_rbf`) is created using the `sns.heatmap` function, with annotations turned on to display the values in each cell. This visualization Figure 8. aids in understanding the model's classification outcomes. The `accuracy_score` function is then applied to compute the accuracy of the RBF SVM model by comparing the predicted labels (`rbf_pred`) with the actual labels from the test set (`y_test`). The resulting accuracy score is "0.9891304347826086", providing an overview of the model's overall performance. Finally, a detailed performance report Figure 9 is generated using the `classification_report` function, which includes key metrics like precision, recall, and F1-score for each class. This report is printed to evaluate the model's effectiveness across all classes thoroughly.

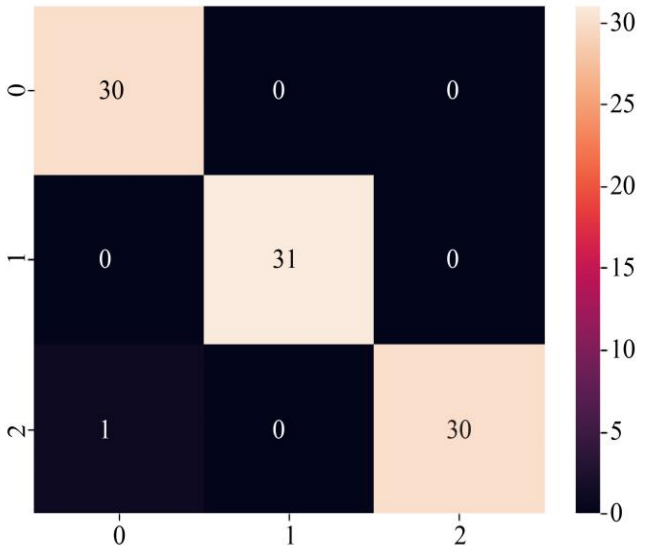

Fig. 8 Confusion matrix for RBF svm

	precision	recall	f1-score	support
0	0.97	1.00	0.98	30
1	1.00	1.00	1.00	31
2	1.00	0.97	0.98	31
accuracy			0.99	92
macro avg	0.99	0.99	0.99	92
weighted avg	0.99	0.99	0.99	92

Fig. 9 Evaluation of RBF svm model

4.4. Sigmoid SVM Algorithm

To evaluate the performance of the sigmoid SVM model, we start by generating a heatmap of the confusion matrix using the `sns.heatmap` function. This heatmap provides a visual representation of the model's classification results Figure 10, with the matrix entries annotated to offer a clear view of the true versus predicted class distributions. The model's accuracy is calculated by comparing the true labels (`y_test`) with the predicted labels (`sig_pred`).

The `accuracy_score` function is "0.40217391304347827" and computes the proportion of correctly classified instances out of the total number of instances, giving a straightforward measure of overall model performance. Finally, a comprehensive classification report is generated using the `classification_report` function in Figure 11, which includes detailed metrics such as precision, recall, and F1-score for each class. This report thoroughly assesses the model's effectiveness across different categories, highlighting its strengths and areas for improvement.

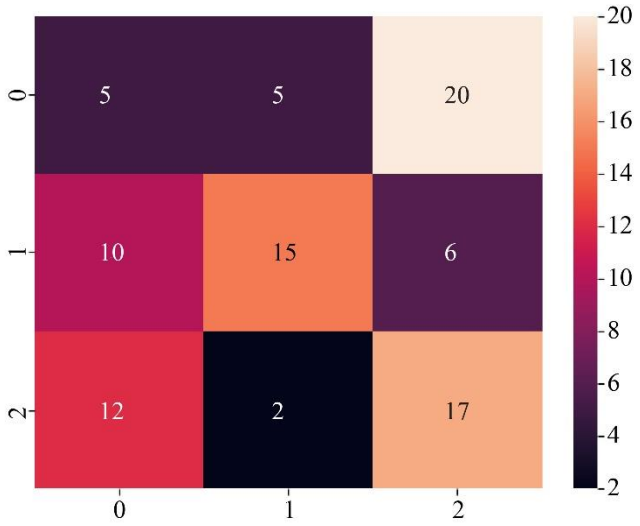

Fig. 10 Confusion matrix for Sigmoid svm

	precision	recall	f1-score	support
0	0.19	0.17	0.18	30
1	0.68	0.48	0.57	31
2	0.40	0.55	0.46	31
accuracy			0.40	92
macro avg	0.42	0.40	0.40	92
weighted avg	0.42	0.40	0.40	92

Fig. 11 Evaluation of Sigmoid svm model

Table 2. Comparative analysis of metrics

Algorithm	Accuracy	F1 score	Precision	Recall
KNN	0.87	0.88	0.85	0.91
Polynomial SVM	0.85	0.85	0.87	0.85
RBF SVM	0.99	0.99	0.99	0.99
Sigmoid SVM	0.40	0.40	0.42	0.40

This approach underscores the importance of selecting an optimal K value and evaluating model performance with relevant metrics to ensure accurate and reliable classification in user profiling. The metrics used to evaluate the model's performance offer a comprehensive assessment of its ability to classify the 'level of content' based on the features extracted from user profiles. These metrics reflect the model's accuracy and effectiveness in distinguishing different content levels. By examining the accuracy, precision, recall, and F1 score values, one can gauge how well the model performs in predicting the correct 'level of content' for various user profiles. High values across these metrics indicate that the model is proficient at making accurate classifications and thus can provide reliable and relevant recommendations. The comparative analysis metrics for the various algorithms are given in Table 2. Based on the metric values obtained, it is determined that the RBF SVM (Radial Basis Function Support Vector Machine) is the best fit for this procedure. This conclusion is drawn from a comparative analysis of performance metrics, where the RBF SVM consistently outperforms other models in terms of accuracy, precision, recall, and F1 score. Consequently, the RBF SVM model is selected for further analysis and module development. The process now involves leveraging the strengths of the RBF SVM to proceed with detailed module specifications and optimizations, ensuring that the model's advantages are fully utilized in the subsequent phases of the project. When the system encounters a user profile whose features do not align with any existing profiles in the database, it recognizes this as a completely new user profile. This situation prompts the system to add the new profile to the User Information Database, thereby updating the system with fresh data for future reference. This addition ensures that the system continues to learn and adapt, incorporating new user profiles into its database to enhance its understanding and improve future recommendations. The revised query provides information that simplifies understanding the user's likely intent and determining the most effective responses to meet their needs.

New Query = implicit query + explicit query

The implicit query spontaneously generates context-aware search queries based on the user's current computing activities. The explicit query enables searchers to advance more efficiently in their search journey than navigating through extensive results lists. This approach provides unique search results and correlating suggestions for the personalized query without relying on extensive and time-consuming patterns. Conversely, when a user profile matches an existing profile, the system retrieves the corresponding profile features from the User Information Database. These features are then mapped to relevant Pattern Model Repository (PMR) information. This process ensures that the query request is highly personalized, integrating information from the user's current query and established profile. By doing so, the system can deliver tailored recommendations and information that address the specific needs and preferences of the user. This personalization enhances the recommendations' relevance and improves the overall user experience by aligning the content more closely with the user's individual context and interests.

5. Conclusion and Future Work

The personalization of query models for search keywords within a corporate context through machine learning techniques significantly enhances the relevance and effectiveness of information retrieval. By employing algorithms such as SVM algorithms and K-Nearest Neighbors (KNN) and evaluating performance through metrics like accuracy, precision, recall, and F1 score, the system adeptly tailors search results to individual users' specific needs and

preferences. This approach streamlines the retrieval process and ensures that the information provided is contextually relevant and aligned with each user's unique profile. The ability to adapt to new user profiles and update the database dynamically ensures that the system remains responsive to evolving requirements. This personalization improves user satisfaction and supports better decision-making within corporate environments by delivering more precise and relevant search results. Overall, integrating machine learning into query personalization represents a significant advancement in optimizing search functionality and enhancing user experience in a corporate setting.Future research in this area could explore several avenues for further development. One important direction is the integration of Questionnaire Evaluation and more advanced machine learning algorithms, such as deep learning models or ensemble methods, to achieve even greater accuracy and nuanced understanding in search personalization. Real-time adaptation to user behaviour and feedback could also be a focal point, ensuring the system remains agile and relevant as user needs evolve. Additionally, incorporating multi-modal data sources, including user interaction logs and contextual information from other corporate systems, may enrich the personalization process, offering a more comprehensive understanding of user preferences. Addressing scalability and performance optimization will be crucial for implementing these systems in large-scale corporate environments while maintaining efficiency and effectiveness.

Equally important is the consideration of user privacy and data security. Ensuring that personalization techniques adhere to data protection regulations and safeguard sensitive information is essential for maintaining user trust. Furthermore, enhancing user experience through intuitive interfaces and mechanisms for collecting user feedback can provide valuable insights for ongoing improvement, contributing to a more responsive and user-centric search system. These future efforts will drive the continued advancement of personalized query models, ultimately delivering more tailored and effective search experiences in corporate contexts.

References

[1] P. Sijin, and H. N. Champa, "Context Based Diversification on Keyword Search by Conceptualization of Typed Terms of the Query," *International Journal of Information Management Data Insights*, vol. 3, no. 2, pp. 1-8, 2003. [CrossRef] [Google Scholar] [Publisher Link]

[2] Lívia Kelebercová, and Michal Munk, "Search Queries Related to COVID-19 Based on Keyword Extraction," *Procedia computer science*, vol. 207, pp. 2618-2627, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] Manish Kumar et al., "Keyword Query Based Focused Web Crawler," *Procedia Computer Science*, vol. 125, pp. 584-590, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[4] Xiangfu Meng, Pan Li, and Xiaoyan Zhang, "A Personalized and Approximated Spatial Keyword Query Approach," *IEEE Access*, vol. 8, pp. 44889-44902, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[5] Seungmin Kim, Eunchan Na, and Seong Baeg Kim, "Developing a Meta-Suggestion Engine for Search Queries," *IEEE Access*, vol. 10, pp. 68513-68520, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[6] Ryen W. White, and Susan T. Dumais, "Characterizing and Predicting Search Engine Switching Behavior," *CIKM '09: Proceedings of the 18th ACM Conference on Information and Knowledge Management*, New York, United States, pp. 87-96, 2009. [CrossRef] [Google Scholar] [Publisher Link]

[7] Bruce Croft, Donald Metzler, Trevor Strohman, *Search Engines: Information Retrieval in Practice*, Pearson Education, pp. 1-552, 2011. [Google Scholar] [Publisher Link]

[8] Ravi Kumar, and Andrew Tomkins, "A Characterization of Online Browsing Behavior," *Proceedings of the 19th International Conference on World Wide Web*, pp. 561-570, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[9] Fernando Diaz, Bhaskar Mitra, and Nick Craswell, "Query Expansion with Locally-Trained Word Embeddings," *Arxiv*, pp. 1-8, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[10] Jaime Teevan, Susan T. Dumais, and Eric Horvitz, "Personalizing Search via Automated Analysis of Interests and Activities" *SIGIR '05: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New York, United States, pp. 449-456, 2005. [CrossRef] [Google Scholar] [Publisher Link]

[11] Ryen W. White, Susan T. Dumais, and Jaime Teevan, "Characterizing the Influence of Domain Expertise on Web Search Behavior," *WSDM '09: Proceedings of the Second ACM International Conference on Web Search and Data Mining*, New York, United States, pp. 132-141, 2009. [CrossRef] [Google Scholar] [Publisher Link]

[12] Marti A. Hearst, *Search User Interfaces*, Cambridge University Press, 2019. [Google Scholar] [Publisher Link]

[13] Eugene Agichtein, Eric Brill, and Susan Dumais, "Improving Web Search Ranking by Incorporating User Behavior Information," *SIGIR '06: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 19-26, 2006. [CrossRef] [Google Scholar] [Publisher Link]

[14] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze, *Introduction to Information Retrieval*, Cambridge University Press, 2008. [Google Scholar] [Publisher Link]
[15] Gerard Salton, and Michael J. McGill, *Introduction to Modern Information Retrieval*, McGraw-Hill, pp. 1-448, 1983. [Google Scholar] [Publisher Link]
[16] Ellen M. Voorhees, "Query Expansion Using Lexical-Semantic Relations" *SIGIR '94*, pp. 67-69, 1994. [CrossRef] [Google Scholar] [Publisher Link]
[17] Claudio Carpineto, and Giovanni Romano, "A Survey of Automatic Query Expansion in Information Retrieval," *ACM Computing Surveys (CSUR)*, vol. 44, no. 1, pp. 1-50, 2012. [CrossRef] [Google Scholar] [Publisher Link]
[18] Ricardo Baeza-Yates, and Berthier Ribeiro-Neto, "*Modern Information Retrieval: The Concepts and Technology behind Search*, Addison-Wesley, pp. 1-913, 2011. [Google Scholar] [Publisher Link]
[19] Fabrizio Sebastiani, "Machine Learning in Automated Text Categorization," *ACM Computing Surveys (CSUR)*, vol. 34, no. 1, pp. 1-47, 2002. [CrossRef] [Google Scholar] [Publisher Link]
[20] Bernard J. Jansen, Amanda Spink, and Tefko Saracevic, "Real Life, Real Users, and Real Needs: A Study and Analysis of User Queries on the Web," *Information Processing and Management*, vol. 36, no. 2, pp. 207-227, 2000. [CrossRef] [Google Scholar] [Publisher Link]
[21] Jacques Savoy, "Statistical Inference in Retrieval Effectiveness Evaluation," *Information Processing and Management*, vol. 33, no. 4, pp. 495-512, 1997. [CrossRef] [Google Scholar] [Publisher Link]
[22] Ryen W. White et al., "Enhancing Personalized Search by Mining and Modeling Task Behavior," *WWW '13: Proceedings of the 22nd International Conference on World Wide Web*, New York, United States, pp. 1411-1420, 2013. [CrossRef] [Google Scholar] [Publisher Link]
[23] Hema Yoganarasimhan, "Search Personalization Using Machine Learning," *Management Science*, vol. 66, no. 3, pp. 1045-1070, 2019. [CrossRef] [Google Scholar] [Publisher Link]
[24] G.D. Zhou, "Recognizing Names in Biomedical Texts Using Mutual Information Independence Model and SVM Plus Sigmoid," *International Journal of Medical Informatics*, vol. 75, no. 6, pp. 456-467, 2006. [CrossRef] [Google Scholar] [Publisher Link]
[25] Cláudia Dias, "Corporate Portals: A Literature Review of a New Concept in Information Management," *International Journal of Information Management*, vol. 21, no. 4, pp. 269-287, 2001. [CrossRef] [Google Scholar] [Publisher Link]
[26] Leif E. Peterson, "K-nearest neighbor," *Scholarpedia*, vol. 4, no. 2, 2009. [CrossRef] [Google Scholar] [Publisher Link]
[27] Elie Raad, Richard Chbeir, and Albert Dipanda, "User Profile Matching in Social Networks," *13th International Conference on Network-Based Information Systems*, Takayama, Japan, pp. 297-304, 2010. [CrossRef] [Google Scholar] [Publisher Link]
[28] Tri Susilowati et al., "Using Profile Matching Method to Employee Position Movement," *International Journal of Pure and Applied Mathematics*, vol. 118, no. 7, pp. 415-423, 2018. [Google Scholar] [Publisher Link]
[29] Xiaodan Yan et al., "A Personalized Search Query Generating Method for Safety-Enhanced Vehicle-To-People Networks," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 6, pp. 5296-5307, 2021. [CrossRef] [Google Scholar] [Publisher Link]
[30] Gloria Chatzopoulou et al., "The QueRIE system for Personalized Query Recommendations," *IEEE Data Engineering Bulletin*, vol. 34, no. 2, pp. 55-60, 2011. [Google Scholar] [Publisher Link]
[31] Alejandro Bellogín, and Pablo Castells, "A Performance Prediction Approach to Enhance Collaborative Filtering Performance," *Advances in Information Retrieval*, pp. 382-393, 2010. [CrossRef] [Google Scholar] [Publisher Link]
[32] Wei Wu et al., "Efficient K-Nearest Neighbor Classification Over Semantically Secure Hybrid Encrypted Cloud Database," *IEEE Access*, vol. 6, pp. 41771-41784, 2018. [CrossRef] [Google Scholar] [Publisher Link]
[33] Karen F. Gracy, "Archival Description and Linked Data: A Preliminary Study of Opportunities and Implementation Challenges," *Archival Science*, vol. 15, pp. 239-294, 2015. [CrossRef] [Google Scholar] [Publisher Link]
[34] Kittipong Chomboon et al., "An Empirical Study of Distance Metrics for K-Nearest Neighbor Algorithm," *The 3rd International Conference on Industrial Application Engineering*, pp. 280-285, 2015. [CrossRef] [Google Scholar] [Publisher Link]
[35] Haneen Arafat Abu Alfeilat et al., "Effects of Distance Measure Choice on K-Nearest Neighbor Classifier Performance: A Review," *Big data*, vol. 7, no. 4, pp. 221-248, 2019. [CrossRef] [Google Scholar] [Publisher Link]
[36] Ding-Xuan Zhou, and Kurt Jetter, "Approximation with Polynomial Kernels and SVM Classifiers," *Advances in Computational Mathematics*, vol. 25, pp. 323-344, 2006. [CrossRef] [Google Scholar] [Publisher Link]
[37] Shunjie Han, Cao Qubo, and Han Meng, "Parameter Selection in SVM with RBF Kernel Function," *World Automation Congress*, pp. 1-4, 2012. [Google Scholar] [Publisher Link]
[38] Quanzhong Liu et al., "Feature Selection for Support Vector Machines with RBF Kernel," *Artificial Intelligence Review*, vol. 36, pp. 99-115, 2011. [CrossRef] [Google Scholar] [Publisher Link]
[39] Hsuan-Tien Lin and Chih-Jen Lin, "A Study on Sigmoid Kernels for SVM and the Training of Non-PSD Kernels by SMO-Type Methods," *Neural Computation*, pp. 1-32, 2003. [Google Scholar] [Publisher Link]