INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2025, VOL. 71, NO. 2, PP. 345–352
Manuscript received February 2, 2025; revised May, 2025. doi: 110.24425/ijet.2025.153579

Learning data characteristics of a session-based recommendation system and their impact on system performance

Urszula Kużelewska

***Abstract*—Recommendation systems are the most effective solution for enhancing user satisfaction and personalising e-commerce services on the internet. These systems use advanced procedures to analyse massive volumes of data, ensuring users receive the most relevant and suitable products available. The success of recommendation systems hinges on the quality of the methods used. However, there is also an impact on the input data. Session-based techniques are the most effective way to generate recommendations. They focus on short-term user interactions organised in sessions. This procedure is the best for real-world scenarios, where one-time users and limited item availability are prevalent. The objective of this study is to examine the relationship between data metrics, including density, shape, and popularity, and the performance of session-based algorithms, in terms of accuracy and coverage.**

***Keywords*—session-based recommenders; evaluation of recommendations; data metrics**

I. Introduction

RECOMMENDER systems are an essential tool for efficiently retrieving relevant information from a vast amount of data. These electronic applications function as digital advisors, collecting behavioural information on users and providing personalised recommendations. Collaborative filtering is the most effective category of recommender systems for providing highly accurate recommendations [1], [2], [3].

In the conceptualisation of collaborative filtering, the user-item matrix includes the users activity over an extended period, taking into account the fact that users have enrolled in the system and have maintained their accounts over time. In realistic situations, such a hypothesis is not always appropriate. Foremost, web applications (e.g. online shops, streaming services) are dominated by one-time or anonymous visitors. No long-term interests are captured for them. In addition, many products (products available for purchase, books or films) are only accessible for a finite duration. Moreover, it is essential to highlight that only novel items are of significant interest to users. [4].

Session-based recommenders concentrate on forecasting the optimal subsequent action for the user within the context of their registered session, which is often anonymous. They take into account the temporal sequence of items in the collected data. They further analyse the current session and other users' past sessions to generate recommendations.

Preference data can vary significantly depending on the domain or even the specifics of the products themselves. Thus, preference data can be described by different characteristics and may affect the performance of the recommendation system. The efficacy of a recommendation system may be contingent upon the context in which it is deployed. In a system with a disparate data structure, for instance, the same system may not perform as well as it would in a different context [5].

This study aimed to determine whether a relationship exists between a session-based recommendation system's data characteristics and performance. The most appropriate data statistics are outlined in [6], [7]. They involve data density, data shape, and item popularity factors: average item popularity and skewness of a long-tail diagram. The accuracy of the system, expressed by a simple HitRate index, was used to measure the relationship. Nevertheless, an assessment was conducted to evaluate the diversity of recommendations with respect to the input data. The tests were conducted on selected methods belonging to different types of systems.

This paper is inspired by the work presented in [8], and [9]. However, the difference resides in the dataset employed in the experimental procedures, different performance metrics utilized to evaluate the recommender, and distinct types of recommenders. In their analysis, the authors of the study [8] focused exclusively on error-based metrics. In contrast, the authors of [9] considered not only error-based metrics but also fairness. However, the dataset used in the conducted experiments was limited to the MovieLens dataset. The experiments performed within this work are based on Diginetica, an e-commerce dataset provided by the company Diginetica [10], with originally 264 thousand ratings, 55 thousand sessions and 32 thousand items. In addition, the diversity metric related to item coverage was investigated.

This article provides the following contributions:

- The characteristics of the input data exert a considerable influence on the accuracy of session-based recommender systems, with the relationship exhibiting notable differences across different types of recommender. Con-

The work was supported by a grant from the Bialystok University of Technology WZ/WI-IIT/3/2023 and funded with resources for research by the Ministry of Education and Science in Poland.

U. Kużelewska is with Faculty of Computer Science, Bialystok University of Technology, Bialystok, Poland (e-mail: u.kuzelewska@pb.edu.pl).

sequently, the performance of these systems should be subjected to individual analysis.
- The accuracy of session-based recommender systems can be enhanced through the implementation of appropriate data preparation techniques for learning data.

This article represents an extended version of a previously published work, which was originally presented in [11]. The extension comprises an analysis of data in relation to statistical values and the performance of recommender systems. Furthermore, the results of an experiment are presented, in which the performance was verified after the removal of the most popular items.

The article is structured into the following sections. The subsequent section presents an investigation of existing literature on the subject matter. The following section, Section III, characterises session-based approach to recommendation generation. Section IV informs the reader of measures employed for evaluation of the recommender systems performance, while Section V outlines the methodology for the calculation of data statistics. Section VI provides the outcomes of the experimental part. Finally, the last section summarises the essential discoveries and provides the conclusions.

II. Related Work

Recent studies have examined the effect of data characteristics on classical recommender systems' performance, providing valuable insights into the correlation between data characteristics and recommendation accuracy [9], [12]. Nevertheless, these studies merely indicate some potential relations and propositions for overcoming the issue without a comprehensive investigation of this topic.

According to Hsu's research [13], skewness has the effect of reducing the precision of collaborative filtering methods. This conclusion was confirmed through experimental findings performed on a real dataset that was naturally skewed. The content of the data was clickstream derived from an online advertisement agency's digital trail.

Shaikh et al. [14] applied a new solution to address data sparsity. They proposed a procedure of data augmentation and cleaning to enhance data characteristics, which in turn improves the accuracy of recommender systems. The procedure involved in data augmentation employs some of the scores that have been predicted with a high degree of confidence in every iteration to augment the training set. Moreover, it also removed the ratings estimated with low confidence from the input data.

The implementation of data management strategies has resulted in notable enhancements to existing methodologies. In the study conducted by Yang [15], the performance of the recommender was significantly in enhanced by incorporating the item's variance minimisation regularisation term, as opposed to the traditional Matrix Factorisation approach.

There are only few recent studies that have extensively examined the effect of data characteristics on classical recommender systems' performance, providing valuable insights into the correlation between data characteristics and recommendation accuracy [9].

The objective of the research [12] was to examine the influence of data statistics on the efficacy of the most predominant shilling attacks against common CF methods. The results demonstrated that data characteristics, in particular size, shape, and density, are significant factors in determining the efficacy of an attack. Moreover, the study identified the most significant features with respect to a specific type of recommender system.

In order to gain insight into the characteristics of the datasets employed in traditional collaborative filtering recommender systems, the authors of [16] conducted a comprehensive literature review. The objective was to identify similarities and differences between these datasets, with the ultimate goal of providing researchers with a set of guidelines to assist them in selecting appropriate datasets for their experiments. The following indices were subjected to investigation: the characteristics of the datasets were examined in terms of shape, space, density, and Gini. The findings indicated that datasets with markedly disparate characteristics enhance the robustness of the evaluation process.

The most recent and comprehensive work [9] proposed an explanatory framework based on regression models to enhance understanding of the impact of data characteristics on the fairness and accuracy of recommender systems. The researchers considered a number of data characteristics, including those related to the structure of the rating matrix and the rating frequency distribution. The results demonstrated that the three most significant characteristics may contribute up to 80–90% towards the overall accuracy of a recommender system. However, no such relationship was evident with regard to the systems' fairness.

III. Session-based Recommender Systems

Session-based methods utilise binary vectors to record users' interactions with items, typically within a defined session. The data is obtained from websites, such as retail outlets or media streaming platforms, and comprises item views, purchases, and listening activities, which are subsequently encoded in the session vectors. The objective of session-based systems is to forecast the subsequent action (item ID) of users.

Session-based recommender systems can be divided into non-neural and neural methods [17]. The first of these work on the similarity between the vector of an active user (the recipient of the recommendations that are generated) and the vectors of its neighbourhood. In this type of methods, the neighbourhood is defined by k objects from the dataset, which are the most similar to the user [18], [19].

The baseline kNN method is IKNN (Item-based kNN), which is discussed in detail in [20]. The method considers solely the most recent item in the active user's session, seeking the item with the highest degree of similarity based on their co-occurrence in other sessions. In this approach, user vectors are converted into binary values, where a value of 1 indicates the occurrence of a particular item in a given session. The similarity between two session vectors is then computed using metrics that are commonly utilized in the RS domain, such as Pearson correlation or cosine similarity [21].

The neural approach employs deep learning methods. The baseline algorithm is GRU4REC set out in [20], which models user sessions using an RNN with Gated Recurrent Units [22] to predict the subsequent item probability for the ongoing session.

In the experiments described in this article, one example of each approach was selected and used for evaluation. The SKNN [19] algorithm was taken as a non-neural technique, while STAMP was considered to be a neural technique. They are presented below.

The SKNN algorithm is designed to analyse all elements present within a given user session, not just the most recent element, and compares it to the other entire sessions with respect to the similarity as well. The set of potential outcomes is restricted to the k most closely related vectors. Subsequently, the item that is present in the majority of neighbouring sessions with the highest degree of similarity to the active user is then presented as the predicted subsequent activity.

In contrast to GRU4REC, STAMP [23] does not apply an RNN. The system employs a short-term attention/memory priority model that draws upon the user's general interests stored in the long-term memory of the current session context, as well as the user's most recent interests retrieved from short-term memory. The general preferences are derived from an external memory constructed from the entirety of historical items accumulated during the course of a session. The attention mechanism is constructed on the embedding of the final item, which represents the user's actual interests.

IV. Evaluation of the Accuracy of Recommender Systems

The evaluation of recommendation generation is conducted through the utilisation of a series of metrics. The most commonly employed metrics for measuring system accuracy are as follows: Hit Rate (HR), Mean Reciprocal Rank (MRR) and Normalised Discounted Cumulative Gain (NDCG) [20].

Furthermore, additional factors are utilised to evaluate the diversity of the generated lists and their capacity to respond to a phenomenon characterised by the predominance of popular items in the recommendations [19].

- **HR** and **MRR**: The calculation of both HR and MRR employs an analysis of the elements present in the generated recommendation lists. Subsequently, the list is truncated at the specified position, and its content is analysed with regard to the presence of items from the test vector. In the case of MRR, the items in the list are associated with the weights, which are inversely related to the order of the items in the test sessions. This methodology is frequently employed in the literature pertaining to the domain of recommendations, for instance [20].
- **NDCG**: The Normalised Discounted Cumulative Gain (NDCG) is a frequently employed metric in the domain of information retrieval. It takes into account the relevance score, i.e. the index of items in the recommendation lists generated by an algorithm, comprising only those items that have been correctly predicted [24].
- **Coverage**: Coverage [25] indicates the frequency with which items occur in the generated recommendations. When the level of Coverage is high, the recommender is able to function correctly. In other words, the recommendations are diverse for different users. Otherwise, there is a tendency to propose a constant set of recommendations to all users, despite their differing taste.
- **Popularity**: The Popularity index, as defined in reference [20], quantifies the extent to which an algorithmic recommendation system exhibits a preference for including only popular items in its recommendation lists. Low values are beneficial in this context, as they indicate the capacity of the methods to address the long tail problem. The index is calculated by taking the average popularity score of the top-k items in the recommendation lists. The final score is obtained by averaging the individual popularity scores of each recommended item. The estimation is derived by counting the occurrences of the items in one of the training sessions and subsequently applying a min-max normalisation process, whereby a score between 0 and 1 is obtained.

In the experimental part of this paper, the baseline metrics to be adopted are as follows: Hit Rate and Coverage, due to the main objective of this research - to examine the relations between statistics of data and the performance of recommender systems. The HitRate was calculated for both recommendation lists, with the length of the lists varying between a short version comprising three elements and a long version comprising 20 items.

V. Calculating Statistical Information from Data

The following section provides an overview of the data statistics and their implications for the overall assessment of the results.

Session-based recommenders are limited to short-term sessions, which consist of a list of items that the user shows interests, unlike classic collaborative filtering systems. The session time is limited to minutes, which means that no historical data is stored for individual users. Any user interactions that occur after the ongoing session has expired are treated as new ones [26]. In the recommender systems which are based on sessions, the data can also be used to create a User Rating Matrix (URM), where the values are binary and indicate whether the user is interested or has not seen the item.

The URM is a matrix comprising columns and rows, with each column corresponding to a system item (V) and each row being associated with a user (U). The configuration of the rating matrix is a pivotal element, delineating the proportion of users to products within the system, as illustrated in Equation 1.

$$Shape(URM) = \frac{|U|}{|V|} \tag{1}$$

A review of the literature reveals instances where the similarity index is contingent upon the Shape value [9]. In cases where the number of users exceeds the number of products, the similarity between users is a more valuable metric. Conversely, when the number of users is less than the number of products, it is more advantageous to utilise the similarity between products [7].

A further pivotal element of the input data is its density, which denotes the ratio of user interests relative to the total matrix size (see Equation 2).

$$Density(URM) = \frac{n_r}{|U| \cdot |V|} \tag{2}$$

where n_r in URM is a number of all ratings.

Low data density is a common issue faced by recommender systems, especially session-based ones. The considerable quantity of processed data and restricted user access to the full range of products within the system, coupled with the ongoing introduction of new products and users, has a detrimental impact on density. Consequently, it is possible that recommendation algorithms may exhibit reduced accuracy when utilising data sets of low density [8], [9].

The influence of popular products on the efficacy of the recommender is a noteworthy attribute [27]. It is anticipated that the products often presented in user sessions, which will ultimately lead to a reduction in system efficiency, a decline in product space coverage, and a narrowing of recommendation lists in terms of diversity [8], [9]. The popularity of a given product is calculated by sorting the products in descending order of their ratings. When this configuration is displayed as a histogram, the diagram reveals a modest number of elevated values at the outset of the series, succeeded by an extensive tail comprising a multitude of products that are seldom rated. A number of techniques may be employed to ascertain the prevalence of popular products within the grade distribution. The average popularity (Equation 3) and the skewness index, which is related to the asymmetry of distribution of data (Equation 4) are suitable examples.

$$AvgPop(URM) = \frac{1}{|U|} \cdot \frac{\sum_{i \in Ru} \phi(i)}{R_u} \tag{3}$$

where R_u refers to items that appear in a user's u session set.

The AvgPop metric calculates the average popularity of items across sessions. An item's popularity score (denoted as $\phi(i)$) is determined by the number of users interacting with it across the entire user set. The averages are calculated over the user's session and then over the set of all sessions.

$$LongTailSkewness(URM) = \frac{1}{|V|} \cdot \frac{\sum_{i=1}^{|V|} (\phi(i) - \mu)^3}{[\frac{1}{|V|} \sum_{i=1}^{|V|} (\phi(i) - \mu)^2]^{\frac{3}{2}}} \tag{4}$$

where μ represents the averaged overall popularity of all items.

The LongTailSkewness coefficient is more sensitive to the actual popularity values with respect to the size of the long tail items. Its larger values indicate the larger size of the long tail.

VI. Experiments

This section presents the outcomes of experiments performed to verify whether the recommendation algorithms performance may be related to data characteristics. The experiments were conducted on 51 subsets of Diginetica dataset [10], specially prepared so as to obtain certain values of the characteristics. Each test involved at least 25 subsets. Selected data examples, with their corresponding statistics, are presented in Table I, with particular attention paid to the range of values. Similarly, in the tables that follow, only a selection of rows from the original lists are presented due to the comprehensive size of the lists and the necessity for visualisation of the relevant outcomes.

A. Datasets

To ensure a comprehensive analysis of the data, multiple subsets with varying characteristics were generated. For each experiment, a unique set of data was selected to consider various aspects. In the first and second experiment, which focus on the Shape and Density of the rating matrix, the objective was to obtain data with disparate proportions of users to products. By providing different numbers of users and items while keeping other statistics constant, different Shape and Density rates were achieved.

TABLE I
INFORMATION DATA ON SELECTED SUBSETS USED IN THE EXPERIMENTS. MINIMAL AND MAXIMAL VALUEAS ARE IN BOLD

Subset	Actions	Sessions	Items /Users	Actions /Users	Actions /Items
ds1-1	165048	27563	35053	**5.99**	4.71
ds1-2	165041	30863	35344	5.34	4.67
ds1-3	165040	**35555**	**35351**	4.64	4.67
ds2-1	**165084**	27612	34932	5.98	4.72
ds2-2	**165084**	27612	34932	5.98	4.72
ds2-3	165041	30934	35266	5.33	4.68
ds9-1	128616	21566	28581	5.96	4.50
ds9-2	129089	24118	28711	5.35	4.50
ds9-3	128975	27790	28785	4.64	4.48
ds12-1	**21786**	**12632**	**5000**	**1.72**	**4.36**
ds12-2	45244	17907	10000	2.53	4.52
ds12-3	67883	20126	15000	3.37	4.52
ds16-1	118620	27654	20000	4.29	5.93
ds16-2	117620	27629	19500	4.26	6.03
ds16-3	116620	27598	19000	4.23	**6.14**

The following experiments were concentrated on various factors of popularity of items. The products were organised in accordance with their respective popularity values and confidently iteratively removed to retain the same total number of users. The obtained sets were subjected to an analysis in relation to the skewness coefficient, and only those sets exhibiting substantial alterations in the coefficient were selected.

Recommendations were obtained using session-based collaborative filtering algorithms (the code of their implementation was received from the Session-Rec framework [28]). The following methods were used: SKNN (a neighbourhood-based approach) with a similarity component utilised cosine measure and STAMP, an algorithm based on a neural network. The generated recommendations have been then evaluated in terms of to the following criteria. The accuracy of the results was evaluated using the HitRate metric, which is a standard measure employed in session-based approaches. The calculation procedure evaluates the composition of the

generated lists when successive items are gradually added to the test sessions. Once the propositions have been generated, the list is truncated at the specified level, and the content is subjected to an examination in accordance with the presence of items derived from the test vector. In this study, short and long thresholds are examined, specifically 3 and 20 elements on the recommendation lists (HR@3 and HR@20).

Coverage [25] indicates the frequency with which items are placed in the recommendation lists. In the majority of cases, the coverage cut-off is equal to 20.

B. Obtained Results

The prepared datasets were employed for the creation and assessment of recommendation lists. Cross-validation was used with a minimum division into 27 sets and results averaging.

The first experiment focused on the shape of the rating matrix. Table II shows the results ordered by the ascending values of the factor. It can be seen that the outcomes for both algorithms are different. For STAMP there are no meaningful positive changes in any of the statistics based on the shape of the matrix. Whereas, for SKNN an increases in accuracy is observed as the Shape value grows.

There is also a correlation between Shape and HitRate - the value is 0.80 (HR@3) and 0.90 (HR@20) - see Table III. In the case of the STAMP algorithm, the correlation is not particularly strong: 0.09 (HR@3) and 0.23 (HR@20). The Coverage measure was unrelated to the Shape value - the correlation in both cases was low (respectively -0.15 and 0.07).

The results confirm the literature's findings that greater accuracy is achieved when Shape values exceed 1. This is because session-based systems are of the user-based type, which means that similarity is calculated among the user's sessions.

The assessment outcomes in the case of Density and the generated recommendation lists are shown in Table IV. As anticipated, no significant correlation between the density and the accuracy of the lists was observed, and the HitRate increases as the data density rises. However, STAMP generated more accurate short list recommendations, while SKNN produced more precise long list recommendations. The correlation indices were as follows. For STAMP, Density and HR@3 was 0.81, while Density and HR@20 was 0.78 (Table III). For SKNN the values were 0.38 and 0.53 respectively. The Coverage measure was unrelated to the Density value in the case of SKNN method (correlation was equal to 0.12). In the STAMP algorithm, the relation was average (correlation was equal to 0.4).

Tables V and VI provide the results of the experiments related to relationship between the recommendation lists and the popularity-based measures: Average Popularity and LongTailSkewness. Although both measures relate to similar characteristics, the outcomes are different. Indeed, a high average popularity of all items has a negative impact on recommendation accuracy. The correlations between AvgPop values and HR@3 are as follows: -0.28 for STAMP and -0.73 for SKNN. The correlation between AvgPop and HR@20 is -0.35 for STAMP and -0.92 for SKNN. Only the long cut recommendations generated by SKNN show sensitivity to popular items in the data, but their accuracy is high (0.9071 to 1.0) regardless of the popularity level. The top 3 recommended items less dependent of this feature for both algorithms, especially for STAMP. Again, the Coverage measure was unrelated to the AvgPop value in the case of SKNN method (correlation was equal to 0.10), however, in the STAMP algorithm, the relation was average (correlation was equal to 0.57).

The LongTailSkewness values have a stronger influence on the accuracy of recommendations. In particular for the SKNN algorithm, the HitRate indices are highly correlated with LongTailSkewness, and the values for HR@3 and HR@20 are 0.87 and 0.97 respectively. As before, the STAMP recommender demonstrates a higher ability to personalise, due to the low correlation values for the HR@3 (0.33) and HR@20 (0.4) indices. The suggestions obtained from the data characterised with high skewness were assessed with a HR@3 of 0.4466 (STAMP) and 0.3281 (SKNN), while the HR@20 was as follows 0.5994 (STAMP) and 0.9262 (SKNN). This indicates that the most popular items in the learning data have a detrimental effect on the performance of the recommendation systems, ultimately leading to the generation of less accurate suggestions. However, SKNN generates precise recommendations despite the skewness of the input data. The Coverage measure was unrelated to the LongTailSkewness value - the correlation in both cases was low (respectively -0.08 and -0.15).

C. The Most Popular Items and Accuracy

The experiment was designed to examine whether there is possibility of enhancement the quality of a recommender system by preparing the input data in an appropriate way. The original datasets were evaluated against the data with the most popular items removed, using the STAMP and SKNN systems. The comparison was based on the same metrics, specifically HR@3, HR@20 and Coverage. Table VII presents the results.

The experiment was conducted on selected sets described by different characteristic values. The performance increased after removing 10% of the most popular items, particularly for the SKNN algorithm, where the improvement is often significant. Specifically, the HR@3 metric increased from 0.3333 to 0.7297, HR@20 increased from 0.5 to 0.8649, and Coverage increased from 0.0033 to 0.0164 in the $ds2-2$ dataset. Considerable similarity can be seen in the $ds2-3$, $ds2-4$ and $ds2-5$ datasets.

VII. Conclusions

Thoroughly examining and preparing the data is crucial to improving the accuracy of recommender systems, as certain data characteristics can significantly impact the functioning of a recommendation system, while others have no apparent impact.

This paper presents experimental findings on the correlation between 4 data characteristics: Shape, Density, Popularity, and LongTailSkewness, and the recommendation algorithms based on session analysis. The obtained outcomes suggest

TABLE II
SHAPE STATISTICS AND RECOMMENDATION PERFORMANCE.

Actions	Sessions	Items	Shape	Performance of recommender systems					
				STAMP			SKNN		
				HR@3	HR@20	Cov.	HR@3	HR@20	Cov.
150481	25848	32483	0.7760	0.6606	0.8307	0.1938	0.1727	0.9131	0.1272
140088	27234	30629	0.8581	0.5577	0.7668	0.2986	0.2559	0.8997	0.2009
132317	30174	28913	0.9985	0.5798	0.7692	0.2843	0.3483	0.8913	0.2038
128247	32946	27991	1.1747	0.6713	0.8825	0.2256	0.2656	0.9222	0.4249
88141	28767	19465	1.4566	0.6981	0.9429	0.3853	0.2877	0.9450	0.5521
125107	44249	26587	1.6943	0.3286	0.5934	0.0993	0.2875	0.9243	0.1025
29637	15234	6667	2.4001	0.7364	0.9494	0.2406	0.4499	0.9845	0.2429

TABLE III
CORRELATION OF DATA CHARACTERISTICS AND RECOMMENDATION ACCURACY.

Name of charact.	Performance of recommender systems					
	STAMP			SKNN		
	HR@3	HR@20	Coverage@20	HR@3	HR@20	Coverage@20
Shape	0.09	0.23	-0.15	0.80	0.90	-0.07
Density	0.81	0.78	0.40	0.38	0.53	0.12
AvgPop	-0.28	-0.35	0.10	-0.73	-0.92	0.57
LongTailSkewness	0.33	0.40	-0.08	0.87	0.97	-0.15

TABLE IV
DENSITY STATISTICS AND RECOMMENDATION PERFORMANCE.

Actions	Sessions	Items	Density	Performance of recommender systems					
				STAMP			SKNN		
				HR@3	HR@20	Cov.	HR@3	HR@20	Cov.
156017	50309	33396	0.00009	0.4166	0.6658	0.0968	0.3600	0.9426	0.2444
156103	36619	33676	0.00013	0.4258	0.5806	0.0662	0.3622	0.8823	0.2325
157838	28952	33876	0.00016	0.6402	0.7796	0.0962	0.2414	0.9468	0.0731
115266	23935	25606	0.00019	0.6820	0.9218	0.4672	0.2002	0.8840	0.3434
105410	21584	23432	0.00021	0.6814	0.9300	0.4584	0.1878	0.8893	0.3237
52564	19144	11667	0.00024	0.7140	0.9466	0.3368	0.3051	0.9505	0.3116
22173	13152	5000	0.00034	0.7228	0.9384	0.2162	0.5029	0.9903	0.2133

TABLE V
POPULARITY STATISTICS AND RECOMMENDATION PERFORMANCE.

Actions	Sessions	Items	AvgPop	Performance of recommender systems					
				STAMP			SKNN		
				HR@3	HR@20	Cov.	HR@3	HR@20	Cov.
159027	46540	34030	16.18	0.4152	0.6022	0.0659	0.3921	0.9180	0.1638
148097	30068	30151	14.45	0.6043	0.7607	0.1119	0.3416	0.9402	0.1793
112257	25572	22463	13.38	0.6189	0.8776	0.4931	0.2088	0.8956	0.3954
77394	19349	17216	12.46	0.6941	0.9329	0.3845	0.2772	0.9203	0.2965
90200	20063	26750	5.72	0.7070	0.9055	0.4656	0.2401	0.9795	0.2952
36395	15449	18750	2.37	0.5656	0.8455	0.1884	0.7187	1.0000	0.0577
17082	10678	12000	1.59	0.5975	0.7846	0.0370	0.9963	1.0000	0.0088

TABLE VI
LONGTAILSKEWNESS STATISTICS AND RECOMMENDATION PERFORMANCE.

Actions	Sessions	Items	LongTail Skewness	Performance of recommender systems					
				STAMP			SKNN		
				HR@3	HR@20	Cov.	HR@3	HR@20	Cov.
165050	38779	35176	-14052	0.4466	0.5994	0.0070	0.3281	0.9262	0.0042
104850	24673	28411	-11313	0.7073	0.9420	0.4000	0.2563	0.9260	0.4353
93895	21552	26000	-10446	0.6991	0.9151	0.4472	0.2363	0.9392	0.3168
74294	20982	18958	-7585	0.5825	0.8430	0.3311	0.5212	0.9527	0.2153
32181	14092	13875	-5552	0.4479	0.7112	0.1442	0.7976	0.9779	0.0920
22900	11923	8625	-3452	0.8072	0.9667	0.1365	0.7116	0.9892	0.1274

TABLE VII
COMPARISON OF RECOMMENDATION ACCURACY AFTER REMOVING THE MOST POPULAR ITEMS. THE SET WITHOUT THE MOST POPULAR ITEMS HAVE A '-REM' SUFFIX. THE IMPROVED VALUES ARE DENOTED IN BOLD.

Dataset	Performance of recommender systems					
	STAMP			SKNN		
	HR@3	HR@20	Cov.	HR@3	HR@20	Cov.
ds2-1	0.1702	0.9149	0.0086	0.6808	0.8723	0.0180
ds2-1-rem	**0.1778**	**0.9555**	0.0084	0.6444	0.8222	0.0176
ds2-2	0.1667	1.0000	0.0021	0.3333	0.5000	0.0033
ds2-2-rem	**0.2432**	0.9730	**0.0076**	**0.7297**	**0.8649**	**0.0164**
ds2-3	0.2917	0.8750	0.0077	0.2083	0.4167	0.0118
ds2-3-rem	0.2500	**1.0000**	0.0071	**0.9062**	**0.9687**	**0.0137**
ds2-4	0.5714	1.0000	0.0046	0.6667	0.8571	0.0077
ds2-4-rem	0.3333	1.0000	**0.0052**	**0.9583**	**1.0000**	**0.0127**
ds2-5	0.1667	0.9167	0.0047	0.1667	0.5000	0.0066
ds2-5-rem	**0.3636**	**1.0000**	**0.0050**	**0.9091**	**1.0000**	**0.0120**

that some algorithms are more strongly correlated than others. STAMP, the neural network-based recommender, demonstrated strong resistance to the data features. Meanwhile, SKNN, the neighbourhood-based system, generated propositions with accuracy related to the data character. Over all, the recommendations of SKNN were significantly more accurate than those of STAMP. SKNN demonstrated a significant improvement after removing the high popularity object from the data.

Examining the data characteristics in recommendation tasks enhances the final algorithm performance. The ability to identify correlated features with the same degree of accuracy as that used to create recommendation lists allows for the preparation and deployment of procedures designed to improve the quality of the data in question.

The results are preliminary and further experimentation is required to develop the assumptions. We must examine further datasets and compare their characteristics. We will investigate further algorithms with respect to their types, for example, neighbourhood-based and neural network approaches. Finally, we will consider further data statistics, including Gini, skewness, and kurtosis of long-tail items.

REFERENCES

[1] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Recommender Systems*. Cambridge University Press, 2010, vol. 24.

[2] F. Ricci, L. Rokach, and B. Shapira, *Recommender Systems: Introduction and Challenges*. Springer, 2015, vol. 1-35.

[3] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, *Collaborative Filtering Recommender Systems*. Springer Berlin Heidelberg, 2007.

[4] D. Jannach, B. Mobasher, and S. Berkovsky, "Research directions in session-based and sequential recommendation," *User Model.-User-Adap. Interaction*, vol. 30, p. 609–616, 2020.

[5] S. Ozdemir and D. Susarla, *Feature Engineering Made Easy: Identify unique features from your dataset in order to build powerful machine learning systems*. Packt Publishing Ltd, 2018.

[6] G. Shani and A. Gunawardana, *Evaluating Recommendation Systems*, 2011, vol. 12, pp. 257–297.

[7] R. Ayub, M. a. Ghazanfar, Z. Mehmood, T. Saba, R. Alharbey, A. Munshi, and M. Alrige, "Modeling user rating preference behavior to improve the performance of the collaborative filtering based recommender systems," *PLOS ONE*, vol. 14, 08 2019.

[8] G. Adomavicius and J. Zhang, "Impact of data characteristics on recommender systems performance," *ACM Trans. Manage. Inf. Syst.*, vol. 3, no. 1, 2012.

[9] Y. Deldjoo, A. Bellogin, and T. Di Noia, "Explaining recommender systems fairness and accuracy through the lens of data characteristics," *Information Processing & Management*, vol. 58, no. 5, p. 102662, 2021.

[10] "Diginetica." [Online]. Available: https://darel13712.github.io/rs_datasets/Datasets/diginetica/

[11] U. Kużelewska and M. Charytanowicz, "Characteristics of the learning data of a session-based recommendation system and their impact on the performance of the system," in *Proceeding of 32nd International Conference on Information Systems Development*, 09 2024.

[12] Y. Deldjoo, T. Di Noia, E. Di Sciascio, and F. A. Merra, "How dataset characteristics affect the robustness of collaborative recommendation models," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '20. Association for Computing Machinery, 2020, p. 951–960.

[13] C.-N. Hsu, H.-H. Chung, and H.-S. Huang, "Mining skewed and sparse transaction data for personalized shopping recommendation," *Machine Learningk*, vol. 57, pp. 35–59, 01 2004.

[14] S. Shaikh, V. R. Kagita, V. Kumar, and A. K. Pujari, "Data augmentation and refinement for recommender system: A semi-supervised approach using maximum margin matrix factorization," *Expert Systems with Applications*, vol. 238, p. 121967, 2024.

[15] W. Yang, S. Fan, and H. Wang, "An item-diversity-based collaborative filtering algorithm to improve the accuracy of recommender system," in *2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, 2018, pp. 106–110.

[16] J. Y. Chin, Y. Chen, and G. Cong, "The datasets dilemma: How much do we really know about recommendation datasets?" ser. WSDM '22. Association for Computing Machinery, 2022, p. 141–149.

[17] M. Ludewig, N. Mauro, S. Latifi, and D. Jannach, "Empirical analysis of session-based recommendation algorithms," *User Model User-Adap Inter*, vol. 31, p. 149–181, 2021.

[18] K. Verstrepen and B. Goethals, "Unifying nearest neighbors collaborative filtering," in *Proceedings of the 8th ACM Conference on Recommender Systems*, ser. RecSys '14. Association for Computing Machinery, 2014, p. 177–184.

[19] M. Ludewig and D. Jannach, "Evaluation of session-based recommendation algorithms," *User Modeling and User-Adapted Interaction*, vol. 28, no. 4–5, p. 331–390, 2018.
[20] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," in *Proceedings International Conference on Learning Representations*, ser. ICLR '16, 2016.
[21] C. C. Aggarwal, *Recommender Systems. The Textbook*. Springer, 2016.
[22] K. Cho, B. Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," p. 1724–1734, 2014.
[23] Q. Liu, Y. Zeng, R. Mokhosi, and H. Zhang, "Stamp: Short-term attention/memory priority model for session-based recommendation," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD '18. Association for Computing Machinery, 2018, p. 1831–1839.
[24] K. Järvelin and J. Kekäläinen, "Cumulated gain-based evaluation of ir techniques," *ACM Trans. Inf. Syst.*, vol. 20, no. 4, p. 422–446, 2002.
[25] G. Adomavicius and Y. Kwon, "Improving aggregate recommendation diversity using ranking-based techniques," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 5, pp. 896–911, 2012.
[26] M. Quadrana, P. Cremonesi, and D. Jannach, "Sequence-aware recommender systems," *ACM Comput. Surv.*, vol. 51, no. 4, 2018.
[27] Smyth, Barry and McClave, Paul, "Similarity vs. diversity," in *Proceedings of the International Conference on Case-Based Reasoning*. Springer, 2001, pp. 347—-361.
[28] "Session-rec software." [Online]. Available: https://github.com/rn5l/session-rec