DECEMBER 2025 Ph.D. in Industrial Engineering ELİF SELEN BABÜROĞLU

REPUBLIC OF TÜRKİYE

GAZİANTEP UNIVERSITY

GRADUATE SCHOOL OF NATURAL & APPLIED SCIENCES

AN ADVANCED CONTEXT-AWARE METHOD FOR DETECTING USER INTEREST DRIFT: METHODOLOGY, STATISTICAL ANALYSIS, AND EMPIRICAL EVALUATION

Ph.D. THESIS

IN

INDUSTRIAL ENGINEERING

BY

ELİF SELEN BABÜROĞLU

DECEMBER 2025

AN ADVANCED CONTEXT-AWARE METHOD FOR DETECTING USER INTEREST DRIFT: METHODOLOGY, STATISTICAL ANALYSIS, AND EMPIRICAL EVALUATION

Ph.D. Thesis

in

Industrial Engineering

Gaziantep University

Supervisor

Prof. Dr. Alptekin DURMUŞOĞLU

Co-Supervisor

Prof. Dr. Türkay DERELİ

by

Elif Selen BABÜROĞLU

December 2025

©2025[Gaziantep University]

AN ADVANCED CONTEXT-AWARE METHOD FOR DETECTING USER INTEREST DRIFT: METHODOLOGY, STATISTICAL ANALYSIS, AND EMPIRICAL EVALUATION

Submitted by **Elif Selen BABÜROĞLU** in partial fulfillment of the requirements for the degree of Doctor of Philosophy in **Industrial Engineering, Gaziantep University,** is approved by,

Prof. Dr. Ahmet Necmeddin YAZICI
Director of the Graduate School of Natural and Applied Sciences …….....

Prof. Dr. Serap ULUSAM SEÇKİNER
Head of the Department of Industrial Engineering …….....

Prof. Dr. Alptekin DURMUŞOĞLU
Supervisor, Industrial Engineering
Samsun University …….....

Prof. Dr. Türkay DERELİ
Co-Supervisor, Industrial Engineering
TEPAV …….....

Exam Date: 26 December 2025

Examining Committee Members:

Prof. Dr. Zeynep Didem UNUTMAZ DURMUŞOĞLU
Samsun University …….....

Prof. Dr. Alptekin DURMUŞOĞLU
Samsun University …….....

Prof. Dr. Eren ÖZCEYLAN
Gaziantep University …….....

Asst. Prof. Dr. Baki ÜNAL
Iskenderun Tehncical University …….....

Asst. Prof. Dr. Yunus EROĞLU
Gaziantep University …….....

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Elif Selen BABÜROĞLU

ABSTRACT

AN ADVANCED CONTEXT-AWARE METHOD FOR DETECTING USER INTEREST DRIFT: METHODOLOGY, STATISTICAL ANALYSIS, AND EMPIRICAL EVALUATION

BABÜROĞLU, Elif Selen
Ph.D. in Industrial Engineering
Supervisor: Prof. Dr. Alptekin DURMUŞOĞLU
Co-Supervisor: Prof. Dr. Türkay DERELİ
December 2025
114 pages

Modern data-driven systems increasingly operate on continuous data streams. In such environments, the statistical properties of data change with external events, contextual factors and system dynamics. This phenomenon, known as concept drift, is lately recognized as a hard-core challenge in learning systems—especially in real-world applications such as recommender systems, and online personalization. User interest drift, an underexplored form of concept drift, is critical in user-centric and decision-critical applications, where delayed or inaccurate adaptation to evolving user behavior might lead to substantial performance degradation. The lack of a statistically principled, context-aware framework in non-stationary environments leaves a significant gap in the literature. Therefore, this thesis introduces C-IDDM (Context-Aware Interest Drift Detection Method), a novel and statistically grounded skeleton for the detection of user interest drift through the explicit integration of contextual information such as *time-of-day*. The proposed method models user behavior derived from contextual text streams and detects preference changes using distribution-free, permutation-based statistical testing. A real-world Last.fm music listening dataset and synthetic streaming datasets with controlled drift scenarios are utilized to evaluate comparative experiments, which indicate that C-IDDM outperforms a context-free baseline and classical streaming drift detectors, particularly under context-dependent drift, achieving lower false positive rates and better context purity.

Key Words: User Interest Drift, Context-Aware Drift Detection, Recommender Systems, Streaming Data, Adaptive Personalization.

ÖZET

KULLANICI İLGİ KAYMASINI TESPİT ETMEK İÇİN GELİŞMİŞ BAĞLAM BİLİNÇLİ BİR YÖNTEM: METODOLOJİ, İSTATİSTİKSEL ANALİZ VE AMPİRİK DEĞERLENDİRME

BABÜROĞLU, Elif Selen
Doktora Tezi, Endüstri Mühendisliği
Danışman: Prof. Dr. Alptekin DURMUŞOĞLU
İkinci Danışman: Prof. Dr. Türkay DERELİ
Aralık 2025
114 sayfa

Modern veri odaklı sistemler, çokça akışkan veriler üzerinde çalışmaktadır. Bu tür ortamlarda, verinin istatistiksel özellikleri dışsal olaylar, bağlamsal faktörler ve sistem dinamiklerine bağlı olarak değişebilmektedir. Literatürde *kavram kayması* olarak adlandırılan bu olgu, öneri sistemleri gibi kontrollü bir ortamın dışında olan sistemler için bir zorluk olarak kabul edilmektedir. Kavram kaymasının görece az incelenmiş bir alt türü olan *kullanıcı ilgi kayması*, kullanıcı merkezli uygulamalarda büyük önem taşımaktadır; kullanıcı davranışındaki değişimlere gecikmeli/ hatalı uyum sağlanması, sistem performansında ciddi düşüşlere yol açabilmektedir. Literatüde bağlamsal bilgiye önem veren istatiksel bir çalışmanın eksiliği öngörülmüş olup, bu eksikliği gidermeyi amaçlayarak, bağlamsal bilgilerin, örneğin; günün saati, açık entegrasyonuna dayanan, istatistiksel temelli yeni bir yöntem olan C-IDDM (Context-Aware Interest Drift Detection Method) önerilmiştir. Önerilen yöntem, bağlamsal metin akışlarından elde edilen kullanıcı davranışlarını dağılımsal temsiller aracılığıyla modellemekte ve kullanıcı tercihlerindeki değişimleri, dağılım varsayımı gerektirmeyen, permütasyon temelli istatistiksel testler yardımıyla tespit etmektedir. Yöntemin etkinliği, Last.fm müzik dinleme veri seti ile kontrollü kayma senaryoları içeren sentetik akış verileri ile değerlendirilmiştir. Elde edilen karşılaştırmalı sonuçlar, C-IDDM'in bağlamdan bağımsız bir temel yönteme ve klasik akış tabanlı kayma algılama yöntemlerine kıyasla, özellikle bağlama bağlı kayma senaryolarında,s daha düşük yanlış alarm oranları ve daha yüksek bağlam saflığı sağladığını göstermektedir.

Anahtar Kelimeler: Kullanıcı İlgi Kayması, Bağlam Farkındalıklı Kayma Algılama, Öneri Sistemleri, Akış Verisi, Uyarlanabilir Kişiselleştirme.

“Dedicated to my one and only,

beloved mother..”

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my supervisor, Prof. Dr. Alptekin DURMUŞOĞLU, and Prof. Dr. Türkay DERELİ for their continuous guidance and endless support throughout my doctoral study. I am thankful for their encouragement, motivation, and belief in me during this journey.

I am also sincerely thankful to my Thesis Monitoring Committee Members, Prof. Dr. Zeynep Didem Unutmaz DURMUŞOĞLU and Assist. Prof. Dr. Baki ÜNAL for their valuable guidance, constructive feedback, and support throughout this thesis. I would also like to thank the committee members of my thesis, Prof. Dr. Eren ÖZCEYLAN, and Assist. Prof. Dr. Yunus EROĞLU for their valuable contributions.

I am sincerely grateful to all my professors who have enlightened my path with their academic knowledge and expertise in the field of Industrial Engineering throughout my undergraduate, master's, and doctoral education.

I would like to express my deepest love and gratitude to my parents; my mom and dad, who could have been the happiest ones if they would have witnessed this moment; my dear brother, worked day and night on my project; my lovely husband; my mother-in-law, and my sister-in-law for their constant encouragement and support, ongoing best wishes.

Finally, I owe my deepest appreciation to my precious twin children, for their endless love and understanding. Without their unwavering support, reaching this point would not have been possible.

Advancing along my academic trajectory brings me immense satisfaction, and I am committed to leveraging forthcoming scholarly pursuits to benefit both my nation and global society.

TABLE OF CONTENTS

LIST OF TABLES

Page

LIST OF FIGURES

Page

CHAPTER 1
INTRODUCTION

1.1 General Remarks

In streaming data, user interest drift poses a crucial challenge, driven by shifts in individual preferences, the pursuit of novelty, external occurrences, cyclical and time-based trends, and situational factors. Current smart systems, such as recommender systems, social media platforms, online advertising, and music streaming services, rely on user interest models to personalize content and maintain engagement. In practice, user behavior is not stable: preferences evolve over time and often vary by context (for example, *time of day* or *situation*). The postulation of constant user interests by a system invariably leads to the consequences of recommending outdated content, overlooking emerging preferences, and generating inconsistent personalization strategies in distinct operational contexts.

This thesis proposes a method designed specifically for contextual, text-derived user signals. The core idea is to represent user interactions as a time-ordered stream, segment that stream into meaningful contexts, and then detect statistically significant changes in the distribution of user-interest representations across time windows. The conduct of individuals is highly contingent upon prevailing circumstances; the same user may exhibit disparate preferences under varying conditions, such as listening to calm music at night and energetic music in the morning, engaging with professional content during work hours and entertainment content afterward, and expressing different opinions depending on situational or social context.

The underlying interests of users are implicit and not directly observable; interactions are often textual or high-dimensional, drift may be gradual and context-specific, and ground-truth drift labels are typically unavailable. The proposed methodology generates a distinct context-aware data stream for each user, converts textual signals into high-density semantic representations through sentence-level embedding. It then exemplifies alterations by means of distributional comparisons, rather than depending on discrepancies in label-

based classification. Nonparametric hypothesis tests are utilized to quantify shifts, a temporal smoothing mechanism integrated to mitigate susceptibility to interference, and multiple statistical indicators combined into a single decision rule that triggers drift alarms when sufficient evidence accumulates.

An evaluation of this strategy is conducted in two mutually reinforcing contexts. Primarily, the study employs synthetic data compilations where drift type, timing, and severity are manipulable, enabling precise measurement of detection accuracy, false alarms, and detection delay. Secondly, this methodology is employed upon an authentic dataset concerning music consumption (Last.fm) to ascertain if context-sensitive drift detection successfully identifies substantial shifts in listening patterns that are overlooked by context-agnostic baseline approaches.

The proposed method is benchmarked against a context-free variant and frequently deployed stream drift detectors, and a sensitivity analysis is conducted to understand how key parameters influence stability and responsiveness. The dissertation furnishes a statistically grounded, context-aware framework for user interest drift detection in streaming environments, along with an empirical evaluation that clarifies when and why contextual factors contribute to superior drift identification accuracy.

1.2 Motivation and Research Questions

Personalized systems have reached a level of maturity where “accuracy” alone is no longer sufficient. Contemporary recommendation engines, social platforms, and music streaming services are expected to continually adapt to users whose interests evolve and vary across diverse contexts. Yet, the dominant assumption underlying many deployed systems, that user preference models remain valid unless explicitly retrained, is fundamentally flawed.

User interests are not static artifacts that can be captured once and reused indefinitely. They are dynamic, context-sensitive, and shaped by both internal factors (taste evolution, novelty-seeking) and external influences (trends, events, time). When systems rely on static or weakly adaptive user models, they inevitably optimize for the *past*, not the present.

This mismatch results in outdated recommendations, reduced user engagement, and an erosion of trust in personalization systems. The problem is not merely that interests change, but in the inherent inability of systems to explicitly identify the occurrence of these changes.

Most personalization pipelines adapt implicitly—through sliding windows, decay factors, or periodic retraining—without addressing a critical inquiry: whether the user's interest distribution has actually shifted in a statistically meaningful way. The lack of this crucial insight dictates that adaptation procedures will develop in a reactive, opaque, and frequently ineffective manner.

The challenge becomes even more severe in contextual environments. User behavior is not uniform but rather systematically divergent depending on contexts such as time of day, situation, or activity. Aggregating all user interactions into a single profile results in the conflation of distinct behavioral patterns into a generalized representation that does not accurately reflect any real user state.Consequently, genuine context-specific changes are either masked or misinterpreted as noise. Disregarding contextual information does not simplify the problem—it introduces structural bias into user modeling.

Furthermore, user interest is frequently expressed through textual signals—queries, content descriptions, comments, or item metadata—rather than explicit labels. In such specific environments, traditional concept drift detection algorithms that depend on classification error rates or numerical attributes are demonstrably ill-suited. The inherent latent, semantic, and distributional characteristics of user interest drift would remain uncaptured through the standard supervised drift detection analysis. These findings highlight a significant discrepancy between theoretical concept drift detection studies and the functional requirements of context-aware, user-centric systems. While the former has produced a rich set of detectors, the latter demands methods that can operate without labels, handle high-dimensional textual representations, and distinguish meaningful behavioral change from contextual variation. This thesis is motivated by the conviction that user interest drift must be detected explicitly, statistically, and contextually, rather than being handled as a side effect of model retraining.

In dynamic systems, achieving reliable personalization fundamentally depends on dri ft detection; it is not merely a supplementary task. Based on this motivation, the study is driven by the following research questions:

RQ1: What methodologies facilitate the identification of evolving user preferences within real-time data streams, circumventing the dependence on pre-labeled results or classifier performance metrics?

RQ2: What is the impact of integrating contextual data on the dependability and comprehensibility of drift detection mechanisms?

RQ3: Is it feasible for statistically rigorous, distribution-aware evaluative measures to reliably indicate drift in multifaceted textual data structures?

RQ4: What is the comparative performance of a context-aware drift detection method against context-free and advanced drift detection techniques with respect to accuracy, occurrences of false alarms, and the latency of detection?

RQ5: How sensitive is the proposed method to key parameters, and under what conditions does it remain stable across different drift scenarios?

1.3 Contributions of the Thesis

In the realm of dynamic, context-aware systems, this work addresses the challenge of user interest drift detection through its innovative methodological, theoretical, and empirical findings. The contributions address fundamental limitations of existing concept drift and user modeling approaches, particularly their inability to distinguish genuine preference evolution from context-induced variability in streaming environments. The main contributions of this thesis are summarized as follows:

- ✓ **A novel context-aware drift detection framework (C-IDDM):**

The first major contribution of this thesis is the explicit formulation of user interest drift as a context-aware, distributional change problem in streaming textual data. Unlike conventional concept drift studies that focus on classifier error rates or numerical feature shifts, this work treats user interest as a *latent semantic distribution* that evolves and varies across contexts. C-IDDM (Context-Aware Interest Drift Detection Method), a unified framework is proposed that explicitly incorporates contextual information into the drift detection process. Unlike context-agnostic

detectors, C-IDDM models user behavior as evolving distributions in latent semantic space and perform context-conditioned comparisons across time, enabling reliable detection of user interest drift under non-stationary conditions.

- ✓ **A statistically grounded, fusion-based detection strategy:**

User interest drift is formulated as a distributional change detection problem in latent semantic space. C-IDDM represents user interactions as sets of semantic embeddings and monitors context-conditioned distributional differences between consecutive time windows. Rather than relying on asymptotic assumptions, the framework employs nonparametric test statistics—including the Kolmogorov–Smirnov distance, the Mann–Whitney U statistic, and mean distance differences—as distributional discrepancy measures**.**

Statistical significance is assessed via permutation-based hypothesis testing**,** yielding empirical p-values that are robust to unknown distributional shapes and embedding-induced dependencies. To suppress spurious fluctuations while remaining sensitive to genuine preference shifts, C-IDDM applies temporal evidence accumulation using an Exponentially Weighted Moving Average (EWMA) over successive permutation p-values. This enables reliable detection of both abrupt and gradual user interest drift.

When multiple statistical signals are available (e.g., across contexts or discrepancy measures), they are combined through p-value fusion using Fisher's method**,** resulting in a single, interpretable drift alarm score with controlled false positive behavior. Together, these components form a statistically principled, assumption-light detection mechanism that achieves robust drift detection under non-stationary and context-rich streaming conditions.

- ✓ **A reproducible benchmarking framework for user interest drift**:

C-IDDM introduces a structured pipeline that constructs context-aware user streams, represents user interests using sentence-level semantic embedding, models temporal change through distance distributions rather than pointwise similarity, and detects drift via statistically grounded hypothesis testing. This framework departs from heuristic or black-box adaptation strategies and instead provides explicit, interpretable drift decisions supported by statistical evidence.

A fully reproducible experimental pipeline was evaluated in this thesis on both controlled synthetic data with known drift properties and a real-world music listening dataset (Last.fm). The framework enables systematic comparison between context-aware and context-free drift detectors, as well as established streaming methods, under consistent evaluation criteria. The evaluation benchmarks C-IDDM against context-free variants and widely used drift detectors, analyzes detection accuracy, false alarm rates, and detection delays, and demonstrates scenarios in which context-awareness yields measurable improvements. This dual evaluation strategy strengthens the validity of the findings and ensures that conclusions are not limited to idealized settings.

- ✓ **Comprehensive sensitivity and robustness analysis:**

An extensive experimental analysis is conducted to assess the impact of key design parameters—such as time bin size, significance thresholds, and temporal smoothing—on detection accuracy, delay, and false positive rates. These results provide practical guidance on the deployment and tuning of context-aware drift detection systems and demonstrate the robustness and generalizability of the proposed approach.

In summary, the study redefines user interest drift as a context-aware, distributional phenomenon in textual streams; proposes a novel, statistically grounded drift detection framework (C-IDDM); integrates hypothesis testing and evidence fusion for robust textual drift detection; bridges concept drift research with practical user interest modeling; and provides extensive empirical validation and sensitivity analysis. These combined findings push the boundaries of existing drift detection techniques and provide a systematic framework for achieving adaptive personalization in evolving systems.

1.4 Structure of Thesis

This thesis is organized to provide a coherent progression from theoretical foundations to methodological development and empirical evaluation. **Chapter 1** introduces the scholarly challenge and underscores the need for context-aware user interest drift detection in dynamic systems. Additionally, specifies the research questions and summarizes the main contributions of the thesis. **Chapter 2** presents the foundational concepts related to streaming data and data stream mining. It systematically

characterizes concept drift, discusses its types, and reviews existing drift detection paradigms, highlighting their assumptions and limitations in user-centric settings.

Chapter 3 reviews the literature on user interest drift detection, with particular emphasis on textual data streams and context-aware modeling. This chapter identifies gaps in existing approaches that motivate the proposed method. **Chapter 4** is dedicated to the articulation of the user interest drift detection problem addressed by this thesis, concurrently introducing the envisioned solution approach. This chapter clarifies the role of context, text representations, and statistical change detection in the overall frameworkand also reveals the proposed Context-Aware Interest Drift Detection Method (C-IDDM) in detail. It describes the construction of context-aware streams, the embedding-based representation of user interests, the statistical drift detection mechanisms, and the final alarm decision process.

Chapter 5 describes the experimental design, datasets, and evaluation metrics used in the study. Both synthetic datasets with controlled drift properties and a real-world music listening dataset are introduced, along with baseline methods for drift detection. **Chapter 6** reports the empirical results and benchmarking analyses. The performance of the proposed method is compared against context-free variants and state-of-the-art drift detectors, focusing on metrics such as Precision, Recall, False Positive Rates, F1 and Context Purity. **Chapter 7** presents a sensitivity and robustness analysis that examines the impact of key parameters on detection performance. This chapter provides insight into the stability and practical applicability of the proposed method. **Chapter 8** discusses the experimental findings and their implications for user interest modeling and adaptive personalization systems. The strengths and limitations of the proposed approach are critically examined. **Chapter 9** concludes the thesis by summarizing the main findings and contributions. It also outlines potential directions for future research in context-aware drift detection and user modeling.

CHAPTER 2
STREAMING DATA, DATA STREAM MINING, CONCEPT DRIFT, AND USER INTEREST DRIFT

2.1 Overview

User interest drift can be formally understood as a specialized instance of concept drift that arises in user-centric, context-dependent, textual data streams. Unlike classical concept drift, which is typically defined with respect to observable target variables and changes in prediction error, user interest drift concerns the latent evolution of a user's semantic preference distribution over time. In this setting, the concept of interest is not directly observable and must be inferred implicitly from user behavior.

Let a user's interaction stream be represented as a sequence of time-ordered observations,where u denotes the user, denotes the contextual state at time denotes a textual signal associated with the interaction, such as a content description, query, or item metadata. User interest drift occurs when the conditional distribution of semantic representations inferred from changes over time for a given user under a given context. Under this formulation, drift is expressed not as a change in observable labels, but as a distributional shift in latent semantic space. This perspective characterizes user interest drift as a context-conditioned, distributional form of concept drift, distinguished by three defining properties. First, the concept of interest is latent rather than explicitly labeled, making direct supervision infeasible. Second, observations are textual and high-dimensional, requiring semantic representations rather than raw feature comparison. Third, behavioral change might seen differently across contexts, such that a global notion of drift may fail to capture meaningful user-level dynamics.

Several challenges naturally arise from this formulation. User behavior in dynamic systems is inherently non-stationary. Preferences evolve due to gradual taste changes, novelty-seeking behavior, and external influences such as trends or events. Consequently, the assumption that historical user data remains representative of future behavior is systematically violated. User interest drift should therefore be regarded not as an anomaly, but as a structural property of user-centric data streams. Simultaneously, user interaction histories are often sparse and irregular. Unlike system-level streams with dense and continuous observations, user-level streams may

contain limited data within short time windows and long periods of inactivity. This sparsity complicates the reliable estimation of user interest distributions and increases sensitivity to noise, rendering naïve drift detection strategies unstable.

A distinct critical challenge is contextual fragmentation. User interests are not globally consistent under all prevailing conditions. The same user may exhibit a particular behavioral pattern depending on contextual factors such as time of day, activity, or situational constraints. Aggregating interactions across contexts disrupts these distinct patterns into a single profile, concealing authentic context-dependent alterations. Accordingly, drift may either remain undetected or be falsely inferred unless context is modeled explicitly.

The user interest drift is rarely accompanied by explicit ground truth. On contrary, in supervised learning scenarios, there are typically no labels indicating when or how a user's interests have changed. Feedback is often implicit, delayed, or absent. This lack of reliable labels precludes error-rate-based drift detection methods and necessitates unsupervised or weakly supervised approaches grounded in statistical evidence rather than predictive performance.

These challenges entail that user interest drift cannot be effectively addressed using traditional concept drift detectors designed for labeled, numeric, and context-free data streams. Instead, detecting user interest drift requires methods that operate on unlabeled, high-dimensional textual representations, explicitly account for contextual variation, and identify change as a statistically significant shift in semantic distributions over time. This study adopts this perspective and develops a context-aware, statistically grounded framework to address these challenges.

2.2 Streaming Data and Data Stream Mining

Recently, evolving data streams have gained more attention than static data streams because evolving streams are encountered repeatedly in real-world problems. Data that are generated continuously at a fast rate from social media, digital marketing, the Internet of Things realms, and hardware devices (such as satellites, sensors, and computers), social media, digital marketing, highway traffic flow monitoring, smartphones, assistive technologies such as Amazon's Alexa, etc., (Azimjonov & Özmen, 2021; Bifet et al., 2010, 2019) comprise a data stream, which is defined as a

sequence of items that arrive in a timely order (Aggarwal, 2007). The environment of data stream application domains includes spam e-mail filtering, climate-change prediction, web log analysis, weather forecasting, fraud detection, sensor networks, intrusion detection, and customer preferences. Streaming data refers to data that is generated continuously over time and arrives sequentially, often at high velocity and potentially without a predefined endpoint.

Such data are unbounded, high-dimensional, and swiftly evolving. In conjunction with the rapidly growing digital universe, the widespread and repeated dissemination of streaming data in many critical real-time tasks has made evolving data streams a hot research topic (Babüroğlu et al., 2021; Ren et al., 2018), growing attention in the machine learning and data mining communities (Santos et al., 2019). Unlike traditional batch datasets, streaming data is characterized by its effectively infinite length, which makes complete storage and repeated access infeasible. As a result, analytical methods must be designed to operate under the assumption that the full dataset will never be available in memory.

Normal activities of everyday life, such as using a credit card or smartphone, often result in the computerised formation of massive evolving data streams. The acceleration of the volume, veracity, velocity, and variety of data streams has triggered the use of machine learning because humans are not capable of analyzing these vast amounts of data. . Learning from these evolving data streams is a challenging task because of not only managing memory and time efficiently but also possible changes in the underlying data distribution (Baena-Garcia et al., 2006; Bifet & Gavaldà, 2007)

Learning depends on the existence of training examples, such as batch/offline learning, and incremental/online learning (Nishida, 2008). Batch learning algorithms are used for batch setting, wherein the data are finite, static, and already pre-processed before the data mining task (Han et al., 2011). In batch/offline learning, the entire set of training data must be available at the beginning. Only once the training is completed, the model can be used for performing predictions.

A defining constraint of streaming environments is the one-pass processing requirement. Each observation is typically processed only once (Hidalgo et al., 2019; Krawczyk et al., 2017), or a very limited number of times, before being discarded or

summarized. This constraint arises from both storage limitations and the need for timely responses. Consequently, learning algorithms must update their internal state incrementally and cannot rely on repeated scans of historical data.

Another fundamental characteristic of streaming data is temporal dependence. Observations are not independent and identically distributed; instead, they are ordered in time, and their statistical properties may evolve. Temporal dependence implies that the relevance of observations often decays over time and that recent data may be more informative than older data. This property directly challenges classical learning assumptions and motivates adaptive modeling strategies. Since the previous data are no longer attainable and unlabeled data is much larger than that of the labeled data (Yin et al., 2020) unsupervised learning should be promoted to discover hidden patterns in streaming data (Elwell & Polikar, 2011; Gözüaçık & Can, 2021; McArthur et al., 2018).

Data Stream Mining (DSM) addresses these characteristics by developing methods that can extract knowledge from streams while respecting resource constraints and temporal dynamics. Foundational DSM concepts include incremental learning, sliding or adaptive windows, synopsis structures, and online statistical monitoring. These techniques aim to balance responsiveness to change with robustness against noise. Despite significant progress, several challenges remain central to DSM. Chief among these is the management of non-stationarity, where the underlying data-generating process changes over time. Additional challenges include memory and computation limitations, delayed or missing labels, and the need to distinguish genuine structural change from short-term fluctuations.

In user-centric applications, these challenges are further amplified by sparse and heterogeneous interaction patterns and the presence of contextual factors that modulate behavior. As a result, effective stream mining in such settings requires methods that are not only computationally efficient but also theoretically grounded in handling evolving distributions over time.

2.3 Concept Drift

A data stream comprises a data set in which the objects have time stamps that induce either a total or partial order between observations, depending on the granularity of the

stamps (Webb et al., 2018). The reason triggered concept drift is the non-stationary and time-variable structure of the underlying data distribution (Schlimmer & Granger, 1986; Widmer & Kubat, 1996). The concept drift between the time points t = 0 and t = 1 can be defined as Equ. (1.1).

$$\exists X : P_{t_0}(X, y) \neq P_{t_1}(X, y) \tag{1.1}$$

where Pt_0 and Pt_1 denote the joint probability distributions at times t_0 and t_1, respectively, for the set of input variables X and the target variable y (class). Changes in the underlying data (concept drift) may be revealed in different forms; the drift may be sudden or abrupt (when the concept is rapidly switching from one to another) or incremental. Concept drift may also occur gradually, indicating that the change in data is not sudden, and may regress incidentally to the previous pattern. Nevertheless, a drift could be accepted as a recurring drift with new concepts that have not been observed before, or previously observed concepts may be repeated after some time. The observed patterns of concept drifts are depicted in Fig. 1.1.

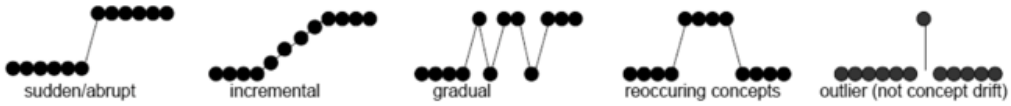

Figure 2.1 Concept drift patterns

Concept drift is a common problem in data stream classification. Therefore, classification algorithms are required to meet four basic requirements to make learning from data streams possible (Gaber et al., 2011; Hulten et al., 2001). For instance, one example can be processed at a time, with the number of scans equal to 1 for each, and only a restricted amount of memory is available for use. The time for performing the work is also limited, and the algorithm must be able to foresee any time at any point.

While the formal definition of concept drift is general, its application to user-centric systems reveals important limitations. In many personalization and recommendation settings, there is no explicit target variable *y* that captures user interest directly. Instead, user preferences are latent and must be inferred from behavioral signals, which are frequently textual, high-dimensional, and context-dependent. Consequently, classical formulations that rely on prediction error or labeled outcomes are insufficient.

The accuracy of classification decreases as concept drifts arise, which emerges that learners should be adaptive to dynamic changes (Shuo Wang et al., 2018). For instance, the users might change their areas of interest or preferences over time in an information system. Hence, learning algorithms employed to form a pattern for the real-world streaming data environment must be able to adapt rapidly and precisely to potential changes (Minku et al., 2010), to lessen the classification error rate.

As a result, adaptation to novel distributions is essential to procure the efficiency of the decision-making process. Adaptive learning indicates the adaptation of predictive models online, in response to unforeseen concept drifts. The algorithms of adaptive learning, called state-of-the-art incremental learning, are adapted to the evolution of the data-generating process over time (Gama et al., 2014). An undesirable deterioration in the performance of the algorithms, such as prediction accuracy, might be originated from the failure of adaptation to drifts (Sixiang Wang & MacHida, 2021).

Drift detection methods, as the principal component of adaptive learning algorithms, are aimed to substitute the base classifier after detecting the drift in the probability distribution of the data improving overall accuracy (Roberto Souto Maior Barros & Santos, 2018; Souto et al., 2019), with minimum delay. Further, the detection algorithms have to abstain from high false positive and false negative rates as they process input data (Sakthithasan et al., 2013). False-positive rates signify false alarms for concept drift, whereas false-negative means missing the real concept drifts out. False-positive entails keeping resources busier, whereas false-negative causes loss in the classification accuracy (Angelov et al., 2008; Bifet & Gavaldà, 2007; Huang et al., 2015).

User interest drift can therefore be understood as a specialized instance of concept drift, where the changing concept corresponds to a user's latent semantic preference distribution rather than an observable label. The drift disloses as a distributional shift in representations derived from user interactions, conditioned on context and time. Moreover, user interest drift often exhibits gradual or incremental patterns and may recur in a context-specific manner, further differentiating it from the abrupt, global drifts commonly studied in synthetic benchmarks. This specialization highlights the need to extend traditional concept drift detection frameworks to accommodate unlabeled data, contextual variation, and semantic representations. Addressing these

requirements is essential for effective drift detection in user-centric, dynamic systems and forms the basis for the approach developed in this thesis.

To address the issue of concept drift, the change has to be detected, which is crucial for trigger-based stream adaptation strategies. Representative user-interest drift methods focus on adaptive learning/ensembles rather than statistically grounded (Hinder et al., 2024), context-conditional drift tests (K. Wang et al., 2024). The most commonly used method for accommodating changes in the underlying distribution of data is the use of drift detectors. The methods can be categorised into two sub-categories based on the dependence on the labelled data: explicit/supervised drift detectors and implicit/unsupervised drift detectors (Sethi & Kantardzic, 2017). Several explicit concept-drift detection methods have been published over the years (Roberto Souto Maior de Barros & Santos, 2019).

2.4 Handling Concept Drift

2.4.1. Base Learners

The **Naïve Bayes** (NB) performs a classic Bayesian prediction under the assumption that all inputs are independent. Bayesian classifiers use statistical theorems to predict the probabilities of class memberships. The NB algorithm is based on the assumption that the values of the attributes (predictors (X)) are independent of each other in the calculation of the probabilities of class P(c) memberships. This assumption is known as the class conditional independence.

The **Perceptron** (P) algorithm is used for binary classification (Freund & Schapire, 1999). The algorithm compares a weighted sum of inputs with a threshold to determine the output. The output is 1 when the summation is greater than or equal to the threshold and 0 otherwise. K-nearest neighbours (KNN), a lazy learner that memorises the training dataset, was introduced by (Silverman & Jones, 1951). To forecast the label of instance x, the classifier investigates the existing instances to determine the k closest instances to instance x. These k training instances are called the KNN of the instance x. The most frequent label among the “k-nearest neighbours” is accepted as the label of instance x. Closeness is characterised in terms of a distance measure, such as the Euclidean distance.

The **decision stump** (DS) models a one-level decision tree, which has a flowchart-like tree structure (Iba & Langley, 1992). A DS generates a prediction by applying the values of an attribute desig decision tree algorithm, was introduced by (Hulten et al., 2001) for learning from extremely large, and possibly infinite, data streams. The HT algorithm mines data streams incrementally without retaining instances in the main memory and can potentially build extremely complex decision tree models with an affordable computa- tional cost. The HT algorithm is based on the observation made by Catlett in 1991 as per which a small subset of the available instances, as a training set, may be sufficient to determine the best attribute as a test node in a decision tree.

Hoeffding option trees (Kirkby, 1994) generate a condensed form of data that act like a set of weighted classifiers, and are con- structed in a cumulative manner similar to that of the regular HTs. The algorithm permits each training example to update a set of option nodes instead of just a single leaf. The option nodes perform in a manner similar to that of standard decision tree nodes, except that they can split the decision paths into various subtrees. Ending the process with an option tree involves the integration of the predictions of the appropriate leaves with the conclusion.nated as the root. The Hoeffding tree (HT) algorithm, also known as the very fast

2.4.2. Concept Drift Detection Methods

Concept drift detectors aim to improve accuracy by locating the positions of concept drifts in large data streams and substitute the base learner according to related changes. The performance of a concept drift detector could be evaluated by considering; a) a number of true positive drift detections, b) number of false alarms, i.e., false positive drift detections, c) drift detection delay, i.e., the time between real drift appearance and its detection. The change detection methods refer to the algorithms that are used to explicitly detect drift points, and distributional changes, in data streams. They characterize and quantify concept drift by discovering the change points or small time intervals during which changes occur (Basseville & Nikiforov, 1993). Concept drift detectors categorized into three general groups (Gama et al., 2014); Sequential Analysis based Methods, Statistical based Approaches and Windows-based Methods.

As statistical based approaches; the drift detection method (**DDM**), proposed by (Gama et al., 2004), detects concept drifts by monitoring the error rate of the classification model. Based on the probably approximately correct learning model

(Mitchell, 1997), in the DDM method, it is assumed that, if the number of examples increases, the error rate of a classifier decreases or remains stable; otherwise, the method claims that there is a drift. Consider *pt* as the error rate of the classifier with a standard deviation of st =√ (*pt* (1 − *pt*)/*t*) at time t. As it processes instances, the DDM updates only two variables *pmin* and *smin* when $pt + st < pmin + smin$. The DDM provides a warning of a drift when $pt + st \geq pmin + 2 * smin$, and detects a drift when $pt + st \geq pmin + 3 * smin$. The values of *pmin* and *smin* are reset when a drift is detected.

In the early drift detection method (**EDDM**), proposed by (Baena-Garcia et al., 2006), the distances between the false predictions for detecting concept drifts is calculated. The EDDM algorithm is based on the observation that drifts are more likely to appear when the errors are closer to each other. The EDDM evaluates the mean interval between two current errors, i.e. *pit*, with its standard deviation *sit* at time t. The method considers the thresholds and searches for a concept drift when a minimum of 30 errors have occurred. The EDDM updates two variables *pimax* and s*imax* when $pit+2 * sit > pimax + 2 * simax$. The method provides a warning of a drift when $(pit+2 * sit)/(pimax+2 * simax)<\alpha$ and indicates that a drift has occurred when $(pit + 2 * sit)/(pimax + 2*simax) < \beta$. In the experiments, the values used for α and β have been set as 0.95 and 0.90, respectively. These values have been determined by the authors after some experimentation. The values of *p*imax and *s*imax are reset only if a drift is detected.

The exponentially weighted moving average **(EWMA)** chart drift detector**,** proposed by (Ross et al., 2012), is a method used for detecting concept drift in streaming classification problems and is dependent on the EWMA chart. EWMA charts are employed for detecting an advancement in the average of a sequence of independent random variables X1, X2,..., Xn, which have common means $\mu 0$ before the alteration point and $\mu 1$ after the alteration point. It is assumed that both the mean and standard deviation ($\mu 0$ *and* σx) of the stream are known. However, in the EWMA for concept drift detection (ECDD), these parameters are not required.

The reactive drift detection method (**RDDM**), proposed by (Roberto S.M. Barros et al., 2017), is based on the DDM; however, it employs an explicit mechanism for disposing off older instances of long-winded concepts to overcome or at least mitigate the performance-loss problem faced by the DDM. In the RDDM, it is assumed that

concepts with thousands of instances create a moderately large number of prediction errors to sufficiently influence the mean error rate and trigger the drifts. Like the DDM, in the RDDM, two variables *pt* and *st* are evaluated over time, and *pmin* and *smin* are updated if $pt + st < pmin + smin$. The constants αw and αd represent the warning and drift levels, respectively. Furthermore, the RDDM grasps three variables: max (the maximum size of a concept), min (the reduced size of a stable concept), and *warnLimit* (the maximum number of instances that limits the warning level).

As sliding window approaches, the adaptive sliding window (**ADWIN**), proposed by (Bifet & Gavaldà, 2007), slides a window "*w"* as the predictions become convenient to detect drifts. In this method, two sub-windows of sufficient widths are operated, i.e., w0 of size n0 and w1 of size n1, where w0.w1 = w. A particular divergence between the averages of the two sub-windows shows a concept drift. Once a drift is detected, the components are removed from the tail of the window until no compelling difference is observed.

SeqDrift2, proposed by (Pears et al., 2014), involves the application of the reservoir sampling method (Vitter, 1985) as an adaptive sampling strategy for random sampling of the input data. In SeqDrift2, entries are collected in two repositories: left and right. As the entries are processed over time, the left repository comprises a combination of older and newer entries selected using the reservoir sampling strategy. The right repository stores the newly arriving entries. Subsequently, an upper bound is discovered for the difference between the means of the two repositories, i.e., μl and μr for the left and right repositories, respectively, using the Bernstein inequality (Bernstein, 1946). Finally, a significant difference between the two averages indicates a concept drift.

SEED, which was proposed by Huang et al., is similar to ADWIN and also involves the benchmarking of two sub-windows within a window *W*. Whenever the two sub-windows of *W (WL and WR)* display distinct averages (higher than a given threshold δ), the older cut of the window (*WL*) is dropped. SEED uses the Hoeffding inequality with the Bonferroni correction proposed in ADWIN for calculating the test statistic Ɛ*cut*. According to the experimental results obtained in a previous study (Huang et al., 2015), it was claimed that the SEED algorithm used for detecting drifts in streams has

a much faster execution time and uses less memory, while exhibiting a false positive (FP) rate, true positive (TP) rate, and detection delay comparable to those of ADWIN2. The parameters of SEED and their respective default values based on the massive online analysis (MOA) framework are a block size (b = 32), a compressed term (c = 75), the threshold (δ = 0.05), a growth parameter that controls the magnitude of the increment in a linear manner (α = 0.8), and the base value for the linear function (ε = 0.01).

DDMs based on Hoeffding's bound (**HDDMA**-test and **HDDMW**-test) were proposed by (Frías-Blanco et al., 2015). In the former, the moving averages are benchmarked for detecting drifts, whereas in the latter, the EMWA forgetting scheme (Ross et al., 2012) is employed to weight the moving averages. The weighted moving averages are then correlated for detecting concept drifts. For both methods, Hoeffding's inequality (Hoeffding, 1963) is used to set an upper bound for the level of divergence between the averages. The authors noted that the first and second methods are ideal for detecting abrupt and gradual drifts, respectively.

The Fast Hoeffding drift detection method (**FHDDM**) (Pesaranghader & Viktor, 2016) uses a sliding window and Hoeffding's inequality (Hoeffding, 1963a,b) for calculating and comparing the maximum (best) probability of the correct predictions observed thus far with the most recent probability of correct predictions for the purpose of drift detection. The authors who presented FHDDM claim that their algorithm provides a lower detection delay, fewer FPs, and fewer false negatives (FNs) in comparison with those of the state-of-the-art detectors. The parameters of FHDDM and their provided default values are the sliding window size ($n = 25$) and probability of error allowed (δ = 0.000001).

In the statistical test of equal proportions (**STEPD**), proposed by (Nishida & Yamauchi, 2007) the accuracy of a chunk C of recent samples is evaluated and correlated with the overall accuracy from the birth of the stream, while employing a chi-square test to check for de- viations. The main idea of STEPD is to oversee the accuracy of a base learner over two windows: a *recent* window containing the latest ex- amples and an *older* window with all the other examples observed by the base learner after the last detected drift. The size of the *recent* window (w) is a parameter, and its default value is 30. Similar to DDM and EDDM, STEPD also has two significance levels for the detection of warnings and drifts: αd = 0.003 and αw = 0.05.

In this detection method, the comparison of the precisions over the two win- dows is conducted using a hypothesis test of equal proportions with a continuity correction, where *rA* is the number of correct predictions in the *nA* examples of the *older* window, *rB* is the number of correct pre- dictions in the *nB* (*w*) examples of the *recent* window, and $p = (rA + rB)/ (nA + nB)$. The method starts detecting a drift after $n \geq 2\ W$ is satisfied, and STEPD stores examples in the short-term memory while $P < \alpha w$ rebuilds the classifier from the stored examples and resets all the variables if $P < \alpha d$. The stored examples are eliminated if $P \geq \alpha w$.

The Fisher exact test drift detector (**FTDD**) (Cabral & Barros, 2018) is a drift detector that is based on the implementation of Fisher's exact test (Fisher, 1922). The authors presented three detectors: the FTDD, Fisher proportions drift detector, and Fisher squared drift detector. The detectors are based on STEPD and the deficiency of its statistical test of equal proportions in small or imbalanced data samples. The Wilcoxon rank sum test drift detector (Roberto Souto Maior de Barros et al., 2018) is also a new generation detector that employs two windows of data for detecting concepts in a manner similar to that of STEPD. Moreover, the WSTD detects drifts based on a skilful implementation of the Wilcoxon rank sum statistical test (Wilcoxon, 1945), whereas in the STEPD, the test of equal proportions is used. WSTD can detect fewer FPs than STEPD and is particularly efficient in detecting abrupt drifts. In addition to the three parameters of STEPD with the same default values, WSTD limits the size of the older window by using an extra parameter (w2 = 4000).

As sequential approaches, the Page–Hinkley test (**PH)**, proposed by (PAGE, 1954), is a sequential analysis technique that is also used for concept drift detection. The test operates and evaluates the actual accuracy values and their mean up to the processed instance. If a drift occurs, the classifier will likely start misclassifying the newly arriving instances, resulting in a decrease in the current and mean accuracy rates. The cumulative (*UT*) and minimum (*mT*) divergences among these two values are evaluated. Higher *UT* values indicate that the observed values vary noticeably from their earlier recorded values. If the magnitude of (*UT –mT)* is larger than a specified threshold (λ), it implies that drift is detected. If the value of λ increases, the result will possibly indicate not only fewer FPs but also more FNs and delays for some de- tections. An extended version of the test was also recently proposed by (Shaker & Lughofer, 2014).

The geometrical moving average (**GMA)**, proposed by (Roberts, 2000), presents the newest observations, the greatest weight, and all previous observation weights decreasing in geometric progression from the most recent back to the earliest. The term "moving average" (without the adjective "geometric") implies an ordinary moving average in which k consecutive points are charged with equal weights of l/k. In the computation of a geometric moving average, the most recent point is assigned a weight of r, where $0 < r \leq 1$, and each point is assigned a fraction (1-r) of the weight of its immediate successor. Tests based on GMAs compare well with numerous run and moving average tests. A wide range of concept drift detection methods has been proposed to identify changes in streaming data. These methods differ in the signals they monitor, the assumptions they make about the data, and the type of drift they are designed to detect. While effective in certain settings, many of these approaches exhibit limitations when applied to user-centric, contextual, and textual data streams.

One of the earliest families of drift detectors originates from statistical process control. Methods such as the Cumulative Sum (**CUSUM**) test and the Exponentially Weighted Moving Average (EWMA) monitor deviations of a monitored statistic from an expected baseline. These techniques are designed to detect shifts in the mean or variance of a process and are valued for their simplicity and low computational cost. However, they typically assume numerical, low-dimensional signals and require careful parameter tuning to balance sensitivity and false alarms. Moreover, when applied to complex data such as text embeddings, the choice of an appropriate summary statistic becomes non-trivial.

Another class of approaches relies on distribution-based statistical tests, including the Kolmogorov–Smirnov (KS) test and the Mann–Whitney U (MWU) test. These non-parametric methods compare samples from different time windows to assess whether they originate from the same distribution. Their appeal lies in their minimal distributional assumptions and applicability to unlabeled data. Nevertheless, their effectiveness depends on sufficient sample sizes and appropriate window selection. In high-dimensional or sparse user-level streams, naive application of these tests may lead to unstable or delayed detections.

Adaptive window methods, such as ADWIN and KSWIN, dynamically adjust window sizes based on detected changes in statistical properties. By maintaining two sub-

windows and testing for significant differences, these methods aim to balance responsiveness and robustness. While adaptive windowing reduces the need for fixed window sizes, these approaches typically assume numeric features and often rely on single-test decisions. Their behavior can be sensitive to parameter choices, and they may struggle to distinguish genuine drift from short-term fluctuations in noisy user behavior.

The Page–Hinkley test is another widely used drift detection method, particularly in online learning settings. It detects changes by monitoring cumulative deviations from a running mean, signaling drift when these deviations exceed a predefined threshold. Although effective for detecting abrupt changes in numerical streams, the method assumes a stable baseline and does not explicitly model distributional change. Its reliance on cumulative error or numeric statistics limits its applicability to unlabeled or semantic data streams.

A common limitation across many existing drift detection methods is their implicit reliance on assumptions that rarely hold in user-centric applications. These include the availability of labeled data, numeric and low-dimensional feature spaces, stationarity within contexts, and uniform drift behavior across users. Furthermore, most methods treat the data stream as context-free, aggregating all observations into a single sequence and thereby obscuring context-specific behavioral patterns.

In contrast, user interest drift in textual data streams is inherently latent, high-dimensional, and context-dependent. Detecting such drift requires methods that can operate on semantic representations, accumulate statistical evidence over time, and distinguish genuine preference evolution from contextual variation and noise. Drift may occur only in certain regions of the input/context space; global detectors may miss it (Liu et al., 2017). These considerations motivate the need for drift detection frameworks that extend beyond traditional detectors, integrating statistical rigor with context-aware modeling, as pursued in this thesis.

2.5. Concept Drift Detection in Textual Data

Detecting concept drift in textual data streams presents challenges that are fundamentally different from those encountered in numerical or categorical domains. Textual data is inherently high-dimensional, semantically rich, and context-sensitive,

making traditional drift detection techniques difficult to apply directly. As a result, drift detection in text requires careful consideration of representation, comparison, and temporal stability.

One of the primary challenges in textual drift detection is high dimensionality. Text representations, whether based on bag-of-words, topic distributions, or neural embedding, typically reside in large feature spaces. In such settings, simple summary statistics may fail to capture meaningful change, while direct comparison of raw features becomes computationally expensive and statistically unstable, particularly in sparse user-level streams.

Another critical challenge lies in distinguishing semantic shift from distributional shift. Changes in word usage or embedding distributions may reflect genuine evolution in user interests, but they may also arise from variations in language, vocabulary, or content availability. Semantic shift refers to changes in meaning or topical focus, whereas distributional shift may occur even when underlying interests remain stable. Drift detection methods that rely solely on surface-level statistics or similarity scores risk conflating these two phenomena.

Recent advances in representation learning have introduced embedding-based approaches for textual drift detection. Neural embeddings map text into dense, continuous vector spaces that preserve semantic relationships, enabling distance-based or similarity-based analysis over time. While embeddings alleviate some challenges of high dimensionality, they introduce new concerns related to embedding instability. Embedding spaces may evolve due to model updates, stochastic training effects, or contextual usage patterns, complicating the interpretation of observed changes.

Prior research on textual drift detection has explored several methodological directions. Early approaches relied on topic modeling, tracking the emergence, disappearance, or evolution of topics over time. Although topic models provide interpretable representations, they often struggle with short texts and require careful tuning of model parameters. More recent studies have employed embedding similarity tracking, monitoring changes in cosine similarity, centroid movement, or clustering structure across time windows. While intuitive, these approaches frequently depend on heuristic thresholds and lack formal statistical guarantees.

A common limitation across these methods is their reliance on pointwise similarity measures or static thresholds, which are sensitive to noise and difficult to generalize across users and contexts. Furthermore, many approaches treat textual streams as context-free, aggregating interactions across situations and thereby obscuring context-specific semantic dynamics.

In summary, concept drift detection in textual data demands methods that can operate in high-dimensional semantic spaces, distinguish meaningful preference evolution from superficial variation, and provide statistically interpretable drift signals. Addressing these challenges requires moving beyond heuristic similarity tracking toward distribution-based, statistically grounded approaches, particularly in user-centric and context-aware settings. These considerations form a central motivation for the framework developed in this thesis.

2.5 Text Mining for User Modeling

Text mining plays a central role in user modeling for modern personalization systems, as user interests are frequently expressed through textual signals such as content descriptions, search queries, item metadata, and user-generated text. Effective modeling of user interests, therefore, depends critically on how textual data is represented and compared over time.

Early approaches to text representation relied on term-based methods, such as bag-of-words and Term Frequency–Inverse Document Frequency (TF–IDF). These representations encode documents as sparse vectors reflecting word usage patterns. While computationally efficient and interpretable, term-based methods suffer from several limitations in user modeling contexts. They ignore semantic relationships between words, are sensitive to vocabulary variation, and scale poorly in high-dimensional spaces. Moreover, they are ill-suited for capturing subtle semantic changes that characterize user interest drift. To address these limitations, distributed text representations, or embeddings, have been widely adopted. Embedding methods map textual units into dense, continuous vector spaces where semantic similarity is reflected by geometric proximity. Models such as Word2Vec and Doc2Vec introduced the notion of learning semantic representations from large corpora, enabling more robust comparison of textual content over time.

More recent advances have led to contextual embedding models, which generate representations that depend on the surrounding context of words and sentences. Among these, **Sentence-BERT (SBERT)** has gained particular attention for user modeling tasks due to its ability to produce semantically meaningful sentence-level embeddings. SBERT enables efficient comparison of short textual units, making it well-suited for modeling user interactions that are often brief and context-dependent.

In the context of user interest modeling, embeddings provide a powerful abstraction by allowing user behavior to be represented as a sequence of vectors in a semantic space. However, modeling user interest solely through point estimates, such as centroids or average embeddings, might complicate important distributional properties. User interests are rarely monolithic; instead, they are better characterized as distributions over semantic space that evolve. This observation motivates distance-based modeling of user interest, where changes are analyzed through the comparison of distributions of distances between embeddings across time windows. Rather than tracking individual embeddings or summary vectors, distance-based approaches capture shifts in the overall structure of semantic representations. Such modeling is particularly well-suited for detecting gradual and context-specific changes, as it emphasizes relative semantic relationships rather than absolute positions.

Distance-based modeling also aligns naturally with statistical drift detection frameworks, enabling the application of non-parametric tests to assess whether observed changes reflect genuine interest evolution or random variation. By combining expressive text representations with distribution-level comparison, text mining becomes a principled foundation for robust user interest drift detection in streaming environments.

2.6 User Interest Drift

User interest drift refers to the temporal evolution of a user's latent preferences, revealed as changes in the semantic characteristics of the content that a user generates or consumes. In contrast to traditional concept drift, which is typically defined in terms of observable input–output relationships, user interest drift concerns latent semantic structures that must be inferred from behavioral data.

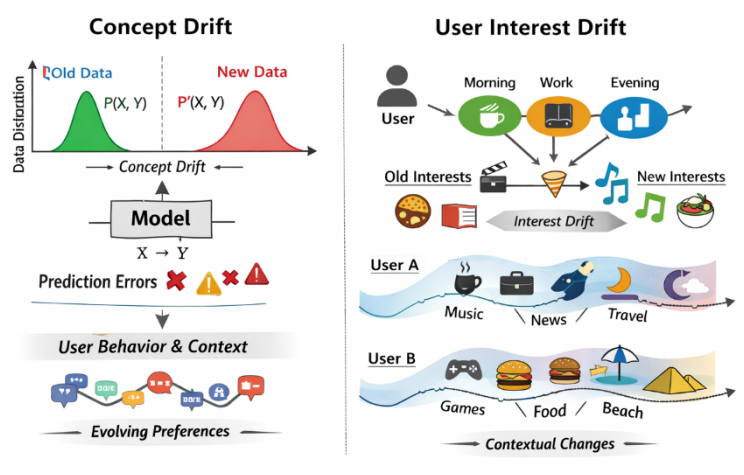

Figure 2.2 Conceptual illustration of concept drift and user interest drift

Formally, user interest drift can be defined as a temporal change in the latent semantic distribution associated with a user's interactions. These interactions are often represented through textual signals, such as item descriptions, queries, tags, or content metadata. Drift occurs when the semantic distribution inferred from these signals changes over time in a manner that reflects evolving preferences rather than random variation.

The study of user interest drift has its roots in recommender systems and personalization research, where adapting to changing user behavior has long been recognized as essential for maintaining relevance. Early recommender systems addressed temporal change implicitly through mechanisms such as time decay, sliding windows, or frequent model retraining. While these approaches allow systems to emphasize recent behavior, they do not explicitly identify when or how user interests have changed.

More recent work has explored explicit modeling of temporal dynamics in user preferences, including time-aware matrix factorization, sequential recommendation models, and recurrent neural architectures. These models incorporate temporal signals to improve predictive performance but often treat preference evolution as a continuous process without identifying discrete or statistically significant change points. As a result, they offer limited interpretability regarding the occurrence and nature of user interest drift.

In parallel, some studies have examined drift through changes in topic distributions, embedding trajectories, or similarity measures over time. Although these approaches provide insights into preference evolution, they frequently rely on heuristic thresholds or domain-specific tuning, making it difficult to generalize results across users and contexts. A common limitation across much of the existing literature is that user interest drift is handled implicitly rather than detected explicitly. Systems adapt continuously but lack a principled mechanism to determine whether observed behavioral changes constitute meaningful shifts in user interests. Moreover, most approaches aggregate interactions across contexts, overlooking the fact that user preferences may evolve differently depending on situational factors.

This thesis adopts the position that explicit detection of user interest drift is both necessary and beneficial. By framing user interest drift as a distributional change in latent semantic space, it becomes possible to apply statistically grounded detection methods that offer interpretability, robustness, and control. This perspective bridges personalization research with concept drift detection, setting the stage for context-aware, statistically principled approaches to adaptive user modeling.

2.7 User Interest Drift Detection Methods

Existing approaches to handling user interest drift in personalized systems can be broadly characterized by how they manage temporal change rather than how they explicitly detect it. In much of the literature, user interest drift is treated as a secondary effect to be mitigated through architectural or algorithmic design choices, rather than as a phenomenon to be identified and analyzed directly.

One common strategy relies on sliding window similarity analysis, where user profiles or representations are constructed over recent interaction windows and compared to past profiles using similarity or distance measures. Changes in similarity over time are interpreted as indicators of evolving preferences. While intuitive, these methods depend heavily on window size selection and similarity thresholds, which are often chosen heuristically. As a result, detection performance is sensitive to parameter settings and difficult to generalize across users with varying activity levels.

Another widely adopted approach involves model retraining triggers based on performance degradation or temporal scheduling. In such systems, user models are

periodically retrained using recent data, or retraining is initiated when predictive performance falls below a predefined threshold. Although effective for maintaining recommendation accuracy, these approaches conflate adaptation with detection. They do not provide explicit evidence of drift, nor do they distinguish between noise-induced fluctuations and genuine preference changes.

In many recommender systems, drift is addressed implicitly through mechanisms such as time-decay weighting, sequential modeling, or continuous online updates. These methods allow models to adapt gradually to new behavior but do not identify discrete change points or quantify the significance of observed shifts. Consequently, the system adapts without awareness of when or why user interests have changed, limiting interpretability and control.

A fundamental limitation of these approaches is the absence of explicit, statistically grounded drift detection mechanisms. Without formal hypothesis testing or significance assessment, it is difficult to determine whether observed changes reflect meaningful user interest evolution or random variation. This limitation is particularly pronounced in settings involving textual data and sparse user interactions, where noise and contextual effects can dominate observed patterns. The lack of explicit detection also impedes comparative evaluation and sensitivity analysis. When drift is handled implicitly, it becomes challenging to assess detection F1 rates, Recall, false alarm rates, or detection delays, as there is no clear decision point to evaluate. As a result, existing methods offer limited insight into the dynamics of user interest evolution.

These shortcomings motivate the need for explicit, statistically grounded user interest drift detection frameworks that operate independently of recommendation models. Such frameworks should be capable of identifying significant changes in user interest distributions, providing interpretable drift signals, and supporting principled adaptation strategies. Addressing this gap is a central objective of the method proposed in this thesis.

2.8 Contextual Text Data and Context Handling

Context plays a fundamental role in shaping user behavior and, consequently, the expression of user interests in textual data streams. In user-centric systems, interactions are rarely independent of situational factors; instead, they are conditioned

on temporal, environmental, and behavioral states. Understanding and modeling this contextual dependency is therefore essential for accurate user interest analysis and drift detection. Context can be broadly categorized into several types. Temporal context captures time-related factors such as time-of-day, day-of-week, or seasonality, which are known to influence user preferences and activity patterns. For example, users may exhibit different content consumption behavior in the morning than at night. Situational context refers to external conditions or circumstances surrounding an interaction, such as location, device, or concurrent activities. Behavioral context captures aspects of the user's recent interaction history or usage patterns, reflecting short-term behavioral states that may modulate preferences.

A context-aware drift detector can explicitly allow contextual shifts (e.g., time-of-day mix) while detecting residual distributional change beyond the permitted context (Cobb & Van Looveren, 2022). In the presence of contextual variation, textual data associated with user interactions reflects not only underlying interests but also the context in which those interests are expressed. Consequently, context handling becomes a critical design decision in user modeling and drift detection. Recent work argues drift monitoring should be context-aware to avoid alarms triggered by expected context variation (Serfilippi et al., 2025).

One common strategy is context collapse, where all interactions are aggregated into a single, context-agnostic representation. This approach simplifies modeling but implicitly assumes that context does not significantly affect user behavior. In practice, context collapse often leads to blurred representations that average over distinct behavioral modes, masking context-specific patterns and reducing the sensitivity of drift detection. An alternative strategy is context-aware segmentation, in which user interactions are partitioned according to contextual states, and separate models or representations are maintained for each segment. This approach preserves context-specific behavior and enables finer-grained analysis of preference evolution. However, it introduces challenges related to data sparsity, as each context-specific stream may contain fewer observations, particularly for infrequent users or rare contexts.

More sophisticated approaches adopt multi-view or multi-model frameworks, where multiple contextual perspectives are modeled jointly. These methods aim to capture interactions between contexts and user behavior, often through ensemble techniques

or joint representations. While powerful, multi-view approaches can be complex to implement and difficult to interpret, and they often require careful coordination between views to avoid overfitting or redundancy. Despite these advances, several gaps remain in existing context handling strategies. Many approaches either ignore context entirely or treat it as a static feature rather than a dynamic conditioning variable. Furthermore, context-aware methods are rarely integrated with explicit drift detection mechanisms. As a result, changes in user behavior that are specific to certain contexts may go undetected or be misinterpreted as global drift.

This thesis addresses these gaps by adopting a context-aware stream construction that preserves contextual distinctions while enabling statistically grounded drift detection within each context. By treating context as a first-class conditioning variable and combining it with distribution-based semantic analysis, the proposed approach aims to detect user interest drift in a manner that is both sensitive to contextual variation and robust to noise.

CHAPTER 3
LITERATURE REVIEW ON USER INTEREST DRIFT DETECTION

3.1 Overview

This chapter reviews existing research on user interest drift detection, with a particular emphasis on textual data streams and context-aware modeling. While concept drift detection has been extensively studied in the data stream mining literature, user interest drift introduces additional challenges related to latent preferences, semantic representations, and contextual variation. The objective of this chapter is threefold. First, it surveys prior work on drift detection in textual data streams. Second, it reviews approaches to modeling and adapting user interests in recommender systems and personalization. Third, it examines how context has been incorporated into user modeling and drift detection. Through this review, the chapter identifies key limitations in existing approaches that motivate the method proposed in this thesis.

3.2 Drift Detection in Textual Data Streams

The current drift detection approaches primarily function in a context-free manner, often fail to capture subtle yet meaningful behavioral changes across contextual conditions. Early research on drift detection primarily focused on numerical or categorical data streams, where changes could be detected through statistical monitoring of feature distributions or classifier error rates. When extended to textual data, these methods faced immediate challenges due to the high dimensionality and sparsity of text representations. Initial approaches relied on term-frequency statistics, detecting drift by monitoring changes in word distributions over time. Although effective for large corpora, these methods were sensitive to vocabulary changes and struggled with short or sparse texts, which are common in user interaction data.

Subsequent work explored topic-based drift detection, tracking changes in topic distributions inferred from probabilistic topic models. Topic evolution provided a higher-level view of semantic change, but topic models often require extensive tuning and perform poorly on short text streams. More recent studies have adopted embedding-based representations, enabling semantic comparison of text across time.

Drift is inferred through changes in embedding similarity, centroid movement, or clustering structure. While embeddings reduce sparsity and capture semantics, many approaches rely on heuristic thresholds and lack formal statistical validation.

3.3 Embedding-Based Approaches and Semantic Drift

The rise of neural language models has significantly influenced textual drift detection. Embedding-based approaches represent documents or interactions as dense vectors in a semantic space, allowing drift to be measured through geometric properties. Several studies track embedding trajectories, analyzing how user or topic representations move over time. Others monitor the decay of similarity between embeddings constructed from different time windows. These methods provide intuitive visualizations and qualitative insights into semantic change.

However, embedding-based drift detection introduces new challenges. Embedding spaces may be unstable due to stochastic training effects or model updates, and 2changes in embeddings do not necessarily correspond to meaningful shifts in user interests. Without statistical testing, it is difficult to distinguish genuine semantic drift from random variation or noise.

3.4 User Interest Drift in Recommender Systems

The progress in recommendation systems where community recognizes drift, but context-specific detection remains limited (Caroprese et al., 2025), underscores the necessity of effectively monitoring dynamic user preferences, and user interest drift has long been acknowledged as a critical challenge. Traditional collaborative filtering models often assume static preferences, prompting the development of time-aware extensions such as time-decayed ratings, sliding windows, and temporal regularization. Many recommender approaches implicitly treat drift via recency, effectively aggregating over other contexts.

More advanced models incorporate temporal dynamics directly, including sequential recommenders and recurrent neural networks. These models capture evolving behavior patterns and often improve predictive performance. However, they typically treat preference evolution as continuous and implicit, without explicitly identifying drift events or change points. As a result, recommender systems adapt to change but rarely detect it. The absence of explicit drift detection limits interpretability, makes

system behavior harder to diagnose, and complicates the evaluation of adaptation strategies.

3.5 Implicit vs. Explicit Drift Handling

A recurring theme in the literature is the distinction between implicit drift handling and explicit drift detection. Implicit methods adapt models continuously through decay mechanisms or online updates, assuming that recent data is more relevant. While effective in practice, implicit methods provide no indication of when a significant change has occurred or whether adaptation is warranted.

In contrast, explicit drift detection aims to identify statistically meaningful changes and trigger adaptation deliberately. In user-centric and textual settings, explicit drift detection remains underexplored. Most existing systems favor implicit adaptation due to its simplicity, despite the lack of interpretability and statistical guarantees.

3.6 Context-Aware User Modeling

Context-aware modeling has been widely studied in personalization research, recognizing that user preferences depend on situational factors. Time-aware recommenders, location-based systems, and activity-aware models all demonstrate the value of incorporating context. Common strategies include contextual feature augmentation, context-specific models, and multi-context representations. These approaches improve recommendation accuracy but are rarely combined with drift detection mechanisms.

In most cases, context is treated as an additional input to prediction models rather than as a conditioning variable for analyzing preference evolution. As a result, context-specific drift patterns remain largely unexplored.

3.7 Context and Drift Detection: Current Limitations

Only a limited number of studies have explicitly examined drift detection in the presence of context. Existing approaches often aggregate data across contexts or assume that drifts are distributed uniformly across all situations. This assumption is problematic in user-centric systems, where preferences may evolve differently depending on context. Context-specific drift may be masked by aggregation or misinterpreted as noise, leading to missed detections or false alarms. Furthermore, few

context-aware approaches employ statistically grounded detection criteria, relying instead on heuristic rules or model performance indicators.

3.8 Identified Research Gaps

The literature reveals several critical gaps that motivate this thesis. First, there is a lack of drift detection methods tailored to textual user interaction streams that provide statistical significance guarantees. Second, existing approaches rarely treat user interest drift as a first-class problem, instead embedding adaptation implicitly within recommendation models. Third, context-aware drift detection remains largely unexplored, despite strong evidence that context shapes user behavior. Finally, there is a limited empirical evaluation comparing context-aware and context-free drift detection under controlled and real-world conditions. Addressing these gaps requires a framework that combines semantic text representations, statistical drift detection (Jeong & Kim, 2023)stati, and explicit context modeling. The method proposed in this thesis is designed to address these limitations directly.

CHAPTER 4
PROPOSED METHOD: C-IDDM FRAMEWORK

4.1 Overview

The preceding chapter established that user interest drift constitutes a distinct and under-addressed exhibition of concept drift that arises in user-centric, contextual, and textual data streams. Unlike classical concept drift, which typically concerns changes in input–output relationships for predictive models, user interest drift reflects the evolution of latent user preferences over time and across contextual conditions. Existing approaches in personalization and recommender systems either adapt implicitly—through decay mechanisms or continuous retraining—without explicitly detecting drift, or rely on heuristic similarity rules that lack statistical rigor and interpretability. As a result, the timing, significance, and nature of preference change often remain opaque.

Within this chapter, the problem of user interest drift detection, which constitutes a central contribution of this thesis, is rigorously formalized and operationalized through a concrete, modular, and interpretable detection pipeline. The chapter introduces the underlying data model for contextual textual interaction streams, formalizes the notion of drift as a distributional change in latent semantic space, and explicitly states the assumptions under which reliable detection is achievable. Together, these elements establish the theoretical foundation of the proposed approach, encompassing the formal problem formulation, guiding design principles, and the high-level structure of the Context-Aware Interest Drift Detection Method (C-IDDM)—a statistically grounded framework designed for detecting user interest drift in dynamic environments.

The central objective of C-IDDM is to identify statistically significant changes in user interests as they evolve over time and across contexts, without relying on labeled outcomes, predictive models, or task-specific loss functions. In contrast to prediction-error–based drift detectors, which infer drift indirectly through model degradation, C-IDDM treats user interest drift as an unsupervised distributional change detection

problem. User behavior is modeled as a sequence of context-conditioned semantic distributions, and drift is defined as a statistically significant change between these distributions across successive time intervals.

This formulation enables direct and interpretable reasoning about preference evolution at the user level, while remaining agnostic to downstream recommendation, ranking, or classification tasks. By decoupling drift detection from model adaptation and task-specific objectives, C-IDDM provides a general-purpose framework that can be integrated into a wide range of user-centric systems. The remainder of this chapter details the problem formulation, design principles, individual components of the framework, illustrating how context-aware stream segmentation, distributional modeling, non-parametric statistical testing, and temporal evidence aggregation collectively instantiate the proposed approach, algorithmic description, and computational considerations.

4.2 Problem Formulation

This dissertation investigates the problem of detecting the phenomenon of user interest drift in continuously changing environments where user behavior is observed through contextual textual interaction streams. Such environments are inherently non-stationary, as user preferences evolve over time due to external events, situational context, and long-term behavioral changes. The primary objective is to analyze statistically significant changes in user interests as they expressed in the semantic content of interactions.

Formally, let U denote the set of users and C the set of contextual variables (e.g., time-of-day). For each user $u \in U$, interactions arrive as a **time-ordered stream** (see Equ. 4.1).

$$S_u = \{(x_i, t_i, c_i)\}_{i=1}^{\infty}, \tag{4.1}$$

where x_i represents a textual signal (e.g., consumed content or activity description) representing the interaction, $c_i \in C$ denotes the associated contextual state associated with the interaction, and t_i is the interaction timestamp of the *i-th* interaction. This formulation explicitly captures the User × Context × Time → Text mapping that characterizes user-centric data streams in personalization systems. Each textual signal

x_i is mapped into a fixed d-dimensional latent semantic embedding space R^d via an embedding function (see Equ. 4.2).

$$\Phi : (x_i, c_i) \rightarrow \mathbf{z}_i \in \mathrm{R}^d , \tag{4.2}$$

which captures both semantic content and contextual information. The resulting embedded stream $\{ \mathbf{z}_i \}$ provides a continuous representation of user interests over time. The embedding function Φ is assumed to be fixed during the drift detection process, ensuring that changes in the embedding distributions are attributable to user behavior rather than representational drift.

User interaction streams are continuous and asynchronous, making direct distributional comparison infeasible. To enable principled analysis under streaming constraints, interactions are segmented into discrete, non-overlapping context-conditioned time bins. Formally, for a given user u and context $c \in C$, the interaction stream is partitioned into successive, non-overlapping time intervals $\{\boldsymbol{B}_{u,c}^{(t)}\}_{t=1}^{T}$, where $T=\{t_1,t_2,\ldots\}$ denote the ordered set of time bins. Each bin aggregates interactions that occur within the same temporal window and share the same contextual condition. This context-aware segmentation ensures that comparisons are performed between behavioral states under comparable conditions, preventing spurious drift detections caused by contextual confounding.

Within each context-conditioned time bin $\boldsymbol{B}_{u,c}^{(t)}$, user interest is modeled as a probability distribution over a latent semantic space. Given an embedding function $\Phi(\cdot)$, each interaction is mapped to a vector $\mathbf{z}_i \in \mathrm{R}^d$. The collection of embeddings in bin t constitutes a finite sample drawn from an unknown probability distribution $P_{u,c}^{(t)}$ representing the user's interest state during that interval over the latent space. This distribution encapsulates the semantic characteristics of user behavior under a specific context and time period. It is significant to note that user interest is not represented by a single vector but by the distributional structure of embeddings within each time bin, which is more robust to noise and short-term variability.

User interest drift is defined as a statistically significant change in the distribution of latent representations across successive time bins under the same context. Specifically, a drift event is said to occur at time bin t if

$$P_{u,c}^{(t)} \neq P_{u,c}^{(t-1)} , \tag{4.3}$$

where the inequality is assessed using a non-parametric statistical test on finite samples from each bin (see Equ. 4.3). By defining drift as a distributional change rather than a pointwise deviation, this formulation explicitly distinguishes persistent preference shifts from transient fluctuations, noise, or isolated anomalies. Under this formulation, user interest drift is not restricted to abrupt global shifts. Instead, the proposed definition naturally accommodates gradual or incremental drift**,** context-dependent drift patterns**,** and user-specific evolution**.** Drift may accumulate slowly across successive time bins, reveal itself individually under distinct contextual conditions, and occur independently for each user stream. This flexibility is essential for modeling real-world behavioral data, where preference changes are neither synchronized across users nor uniform across contexts. It also avoids reliance on labeled outcomes or prediction errors, making it suitable for unsupervised, user-centric settings.

The objective of the drift detection task is to identify the onset of user interest drift as early and reliably as possible, while maintaining control over false positive detections. Formally, given samples from consecutive context-conditioned time bins, the task is to determine whether the null hypothesis

$$H_0: \quad P_{u,c}^{(t)} = P_{u,c}^{(t-1)} \tag{4.4}$$

should be rejected in favor of the alternative hypothesis

$$H_1: \quad P_{u,c}^{(t)} \neq P_{u,c}^{(t-1)}. \tag{4.5}$$

This must be achieved under the constraints of streaming data, limited sample sizes, high-dimensional embeddings, and unknown distributional forms. The detection mechanism should therefore be distribution-free**,** computationally efficient, and capable of accumulating evidence over time to balance detection delay against false alarm rates. Under this formulation, user interest drift detection is framed as a context-aware, distributional change detection problem over temporally segmented latent representations. This principled abstraction provides the theoretical foundation for the proposed C-IDDM framework and directly motivates the use of permutation-based statistical testing and temporal evidence aggregation. Furthermore, the problem is complicated by three key challenges:

(i) *Streaming constraints*, requiring bounded memory and online or quasi-online processing;

(ii) *Context dependence*, where preference evolution may differ across contextual conditions; and
(iii) *High-dimensional representations*, which render classical parametric drift detection techniques ineffective. Accordingly, the task reduces to designing a statistically grounded, distribution-free drift detection mechanism that operates on contextual embedding distributions, accumulates evidence over time, and produces reliable drift alarms with controlled false positive behavior. This formulation serves as the foundation for the proposed C-IDDM framework, which explicitly integrates context, temporal structure, and non-parametric statistical testing to detect user interest drift in real-world data streams.

4.3 Design Principles of the Proposed Approach

The formulation presented in Section 4.2 conceptualizes user interest drift detection as a context-aware, distributional change detection problem in latent semantic space. Building on this formulation, the proposed approach is guided by a set of principled design considerations that directly address the theoretical and methodological challenges identified in earlier chapters. These principles ensure that the resulting framework is statistically grounded, context-sensitive, and suitable for deployment in real-world streaming environments.

A central design principle of this work is the explicit separation between drift detection and adaptation. While many existing personalization and recommendation systems continuously adapt through decay mechanisms or frequent retraining, they typically do so without explicitly identifying when a meaningful change in user interest has occurred. In contrast, the proposed approach treats drift detection as a first-class statistical decision problem, producing explicit drift alarms supported by quantitative evidence. This separation enhances interpretability, enables controlled adaptation strategies, and allows for systematic evaluation of detection performance independently of downstream adaptation mechanisms.

Rather than treating context as an auxiliary or secondary feature, the proposed method models context as a conditioning variable that defines distinct behavioral regimes. User interactions are therefore analyzed within context-specific streams, allowing drift to be detected independently across contextual conditions. This design choice directly

addresses the problem of contextual fragmentation, ensuring that context-dependent interest evolution is not diminished by the synthesis of data from varied circumstances.

User interests are represented as distributions over semantic embeddings rather than as point estimates or static user profiles. This distributional representation captures the inherent variability and uncertainty in user behavior and enables robust comparison across successive time bins. By modeling change at the distribution level, the method is sensitive to both gradual and abrupt shifts in semantic content, while remaining resilient to noise and sparsity in individual interactions. To operationalize this formulation, the proposed approach employs non-parametric statistical hypothesis tests to compare embedding distributions across time bins. These tests are selected to minimize assumptions about the underlying data-generating process, making them well-suited to high-dimensional, unlabeled textual data streams. As a result, drift decisions are grounded in formal statistical criteria rather than heuristic thresholds, improving both robustness and interpretability.

User behavior streams are inherently noisy, and reliance on single-window comparisons may lead to unstable or spurious detections. To mitigate this issue, the proposed framework incorporates temporal evidence accumulation and smoothing mechanisms, enabling drift decisions to reflect sustained distributional change rather than transient fluctuations. This principle balances sensitivity to genuine interest evolution with resistance to false alarms. The framework is further designed to be modular, allowing key components—including embedding models, statistical tests, and temporal aggregation strategies—to be replaced or extended without altering the overall structure of the method. This modularity facilitates reproducible experimentation, benchmarking against alternative techniques, and adaptation to different domains or data characteristics.

Finally, the proposed approach is developed with practical deployment constraints in mind. It operates in an online or quasi-online manner, requires no labeled data, and imposes bounded computational and memory overhead per user and context. These considerations ensure that the method is applicable to real-world systems where scalability, latency, and interpretability are essential.

4.4 High-Level Architecture of C-IDDM Framework

The Context-Aware Interest Drift Detection Method (C-IDDM) is organized as a modular pipeline that transforms raw contextual textual interaction streams into statistically grounded decisions regarding drift. The architecture is designed to operate in an online or quasi-online setting, with bounded memory requirements and independent processing at the user and context levels. At the input layer, C-IDDM receives a continuous stream of user interactions, where each interaction is associated with a timestamp, a user identifier, and contextual attributes. These interactions may correspond to textual content, behavioral descriptions, or metadata-derived text representations. Contextual variables, such as time of day, are treated as first-class components of the data stream rather than auxiliary features.

C-IDDM consists of five main stages, each corresponding to a distinct conceptual and statistical operation within the detection pipeline. In the first processing stage, incoming user interaction streams are segmented by context and discretized into successive time bins, preserving both temporal ordering and situational distinctions in user behavior. This segmentation step defines the temporal resolution at which drift is assessed and ensures that comparisons are conducted between interaction sets generated under comparable contextual conditions. By isolating context-specific behavioral regimes, the framework prevents context-induced variation from being misinterpreted as user interest drift.

Second, the textual signals associated with user interactions—such as item descriptions, tags, or metadata-derived text—are mapped into a shared latent semantic embedding space using a sentence-level encoder. This semantic representation enables meaningful comparison of user behavior across time, even when the surface-level items, vocabulary, or content sources change. The resulting set of embeddings within each time bin constitutes an empirical representation of the user's interest state during that interval.

In the subsequent stage, C-IDDM models user interest as an empirical distribution over embeddings for each user, context, and time bin. Rather than summarizing user behavior through point estimates or aggregated statistics, the framework retains the full distributional structure of the embedded interactions. This representation enables the detection of both subtle and pronounced shifts in semantic content across time.

Fourth, distributional change detection is performed by statistically comparing embedding distributions across successive time bins under identical contextual conditions. C-IDDM employs non-parametric, permutation-based hypothesis testing to assess whether observed differences reflect statistically significant drift, thereby avoiding assumptions about the underlying data-generating process and ensuring robustness in high-dimensional semantic spaces.

Finally, to mitigate noise and instability arising from limited sample sizes and short-term fluctuations, the framework incorporates temporal evidence accumulation. Statistical test outcomes are accumulated and fused over time, enabling the detection of sustained preference shifts while controlling false positive alarms. The output of this stage is an explicit, interpretable drift signal at the user–context level, which can be consumed by downstream systems for adaptation, monitoring, or analysis.

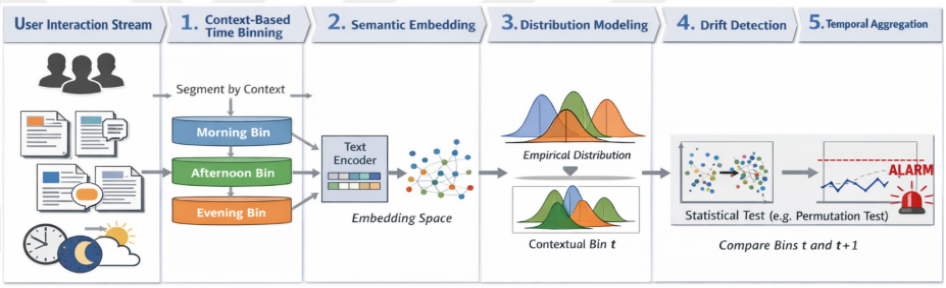

Figure 4.1 High-level architecture of the C-IDDM framework.

4.4.1. Distribution-Based and Statistically Interpretable Design

A defining characteristic of C-IDDM is its distribution-based and statistically interpretable design. Rather than tracking pointwise similarity measures or relying on heuristic thresholds, the method explicitly compares empirical distributions of semantic representations and grounds all detection decisions in formal statistical hypothesis testing. Consequently, detected drift events correspond to statistically significant changes in user behavior, as opposed to opaque or model-dependent signals whose meaning may be difficult to interpret.

This design choice yields several important advantages. First, it improves robustness to noise and outliers, since distributional comparisons are inherently less sensitive to isolated anomalous interactions than pointwise or centroid-based methods. Second, it enables principled operation under sparse or irregular user histories, where predictive

models or error-based detectors may be unreliable, unstable, or infeasible due to limited supervision.

Third, it enhances interpretability and diagnostic transparency, as each drift alarm can be directly traced to a statistically significant distributional shift occurring within a specific context and time interval. By framing user interest drift detection as a distributional hypothesis testing problem, C-IDDM provides a clear semantic interpretation of detected changes and ensures that drift decisions are supported by quantitative statistical evidence. This interpretability is particularly important in user-centric systems, where understanding *when*, *where*, and *under which conditions* preferences change is as critical as detecting change itself.

4.4.2. Construction of Context-Aware Streams and Text Embedding

C-IDDM operates on continuous streams of user interaction data, where each interaction is characterized by a user identifier, a timestamp, and textual or semantic content describing the consumed item or activity. To distinguish genuine user interest evolution from contextual variability, each interaction is additionally annotated with one or more context variables, such as time-of-day or usage condition. These contextual attributes may be directly observed or deterministically derived from temporal information.

The incoming interaction stream is discretized into consecutive, non-overlapping time bins of fixed duration (e.g., daily or weekly), defining the temporal resolution at which drift is assessed. Within each time bin, interactions are further partitioned by context, yielding a collection of context-aware substreams indexed by user, context, and time. Each substream therefore captures the behavior of a specific user under a specific contextual condition during a well-defined temporal interval. For each context-aware substream, the textual content associated with user interactions is mapped into a latent semantic embedding space using a pretrained representation model. This embedding step transforms heterogeneous textual signals into fixed-dimensional vector representations, enabling meaningful comparison of user behavior across time even when surface-level items, vocabulary, or content sources change. By operating in a shared semantic space, the framework abstracts away lexical variation while preserving semantic structure.

As a result, user behavior within each context-conditioned time bin is represented as a set of embedding vectors, rather than as a single aggregated profile or summary statistic. This representation preserves intra-bin variability and provides a robust foundation for subsequent distributional modeling and statistical comparison, which form the core of the C-IDDM drift detection mechanism.

4.4.3. Distance-Based Distributional Modeling and Statistical Testing

To characterize changes in user interests over time, C-IDDM models each context-specific time bin as a distribution in latent semantic space. Rather than directly comparing high-dimensional embedding sets, the framework constructs a compact and robust distributional summary by representing user behavior through distances to a bin-specific centroid. This strategy reduces dimensionality while preserving meaningful information about the internal structure and dispersion of semantic representations.

Formally, for a given user $u \in U$, context $c \in C$, and time bin t, let

$$Z_{u,c}^{(t)} = \{z_1, \ldots., z_{n_{u,c,t}}\} \subset R^d , \tag{4.6}$$

denote the set of semantic embedding vectors corresponding to the user's interactions within that bin, where d is the embedding dimensionality and $n_{u,c,t}$ is the number of observed interactions. For each context-aware time bin, C-IDDM computes a centroid vector

$$\mu_{u,c}^{(t)} = \frac{1}{n_{u,c,t}} \sum_{i=1}^{n_{u,c,t}} z_i, \tag{4.7}$$

which captures the central tendency of the user's semantic behavior while the resulting distance vector captures the dispersion and internal structure of that behavior. Changes in user preferences are shown as shifts in the distribution of distances across consecutive time bins. Accordingly, user interest drift is formulated as a two-sample distributional change detection problem between distance distributions associated with successive temporal intervals. To determine whether an observed change is statistically significant, C-IDDM employs non-parametric statistical tests as discrepancy measures, including statistics based on differences in central tendency as well as rank-based and distribution-based criteria.

Crucially, statistical significance is assessed using permutation-based hypothesis testing rather than relying on asymptotic reference distributions. In each test, embedding vectors from the reference and current time bins are pooled and randomly reassigned into two groups of equal sizes. For each permutation, centroids and corresponding distance distributions are recomputed, and the discrepancy statistic is recalculated. The empirical p-value is then obtained as the proportion of permuted statistics that exceed the observed value.

This recompute-under-permutation strategy yields distribution-free inference and explicitly accounts for dependencies introduced by centroid estimation and latent-space geometry. As a result, the proposed testing procedure remains valid for high-dimensional, non-Gaussian semantic data, making it well-suited to the characteristics of contextual textual interaction streams. To characterize the internal structure of the semantic distribution, each embedding is mapped to a scalar distance from the centroid:

$$d_i^{(t)} = \delta\left(z_i, \boldsymbol{\mu}_{u,c}^{(t)}\right), \tag{4.8}$$

where $\delta(\cdot,\cdot)$ denotes a distance metric in latent space (e.g., Euclidean or cosine distance). The resulting distance distribution:

$$D_{u,c}^{(t)} = \left\{d_1^{(t)}, \dots\dots, d_{n_{u,c,t}}^{(t)}\right\}, \tag{4.9}$$

serves as a compact, one-dimensional summary of user behavior within the given context and time bin. Distributional differences are quantified using a non-parametric discrepancy statistic. Statistical significance is assessed via recompute-under-permutation testing, producing an empirical p-value that reflects the likelihood of observing the measured discrepancy under the null hypothesis of no drift. User interest drift under context *c* is formulated as a two-sample hypothesis testing problem between consecutive time bins t−1 and t. The null hypothesis is defined as

$$H_0: \quad D_{u,c}^{(t-1)} = D_{u,c}^{(t)} \tag{4.10}$$

indicating no change in the underlying semantic distribution, while the alternative hypothesis;

$$H_1: D_{u,c}^{(t-1)} \neq D_{u,c}^{(t)} \tag{4.11}$$

indicates the presence of user interest drift. To quantify distributional discrepancy, C-IDDM employs non-parametric test statistics T(·,·), such as differences in central tendency, rank-based measures, or supremum distances between empirical distributions;

$$T = \left(D_{u,c}^{(t-1)}, D_{u,c}^{(t)}\right), \tag{4.12}$$

These statistics are used solely as discrepancy measures, without reliance on asymptotic reference distributions. Statistical significance is assessed using recompute-under-permutation hypothesis testing, which yields an empirical *p*-value without distributional assumptions. Let

$$T_{obs} = \left(D_{u,c}^{(t-1)}, D_{u,c}^{(t)}\right), \tag{4.13}$$

denote the observed discrepancy statistic. To generate the null distribution, embeddings from the two consecutive bins are pooled:

$$Z_{pool} = Z_{u,c}^{(t-1)} \cup Z_{u,c}^{(t)}, \tag{4.14}$$

and randomly partitioned into two groups of sizes $n_{u,c,t-1}$, and $n_{u,c,t}$. For each permutation b=1,…,B, centroids and distance distributions are recomputed, yielding a permuted statistic $T^{(b)}$. The empirical p-value is then computed as,

$$\rho = \frac{1}{B}\sum_{b=1}^{B} \text{II}\left(T^{(b)} \geq T_{obs}\right), \tag{4.15}$$

where II(·) denotes the indicator function. This recompute-under-permutation procedure explicitly accounts for dependencies induced by centroid estimation and latent-space geometry, ensuring valid inference for high-dimensional, non-Gaussian semantic data. As a result, detected drift events correspond to statistically meaningful changes in user interest structure rather than artifacts of embedding noise or heuristic thresholds.

4.4.4. Temporal Evidence Accumulation and Statistical Signal Fusion

While single-window statistical tests provide localized evidence of change, user interest drift often displays as a persistent temporal process rather than an isolated event. Consequently, drift detection based solely on individual hypothesis tests may be sensitive to noise, sparse observations, or short-term variability. To address this challenge, C-IDDM incorporates temporal evidence accumulation through an

Exponentially Weighted Moving Average (EWMA) mechanism. At each time bin, permutation-based hypothesis testing yields a *p*-value that quantifies the statistical significance of distributional change under a given context. To enable temporal integration, these p-values are first transformed into a continuous evidence signal:

$$e_{u,c}^{(t)} = -\log\left(\rho_{u,c}^{(t)} + \varepsilon\right), \tag{4.16}$$

where $\rho_{u,c}^{(t)}$ denotes the permutation-based *p*-value for user *u* under context *c* at time bin *t*, and $\varepsilon > 0$ is a small constant introduced to ensure numerical stability. The resulting evidence signal is then accumulated over time using EWMA smoothing:

$$S_{u,c}^{(t)} = \lambda\, e_{u,c}^{(t)} + (1-\lambda) S_{u,c}^{(t-1)}, \tag{4.17}$$

where $\lambda \in (0,1)$ controls the trade-off between responsiveness to recent changes and robustness to noise. Larger values of λ favor rapid detection of abrupt drift, while smaller values emphasize long-term accumulation, enabling the detection of gradual or incremental preference shifts. This formulation allows weak but persistent signals to accumulate into a strong drift indication, enabling detection of gradual drift while suppressing isolated fluctuations that could otherwise result in false alarms. Since drift detection is performed independently for each context, multiple context-specific evidence signals may be available at a given time bin. To obtain a unified decision for each user, C-IDDM aggregates context-specific statistical evidence using principled p-value fusion. Let C_t denote the set of contexts for which valid p-values are available at time *t*. Using Fisher's method, the fused test statistic is computed as

$$X_u^2(t) = -2\sum_{c\in C_t} \log\left(\rho_{u,c}^{(t)}\right), \tag{4.18}$$

which, under the assumption of independence, follows a chi-squared distribution with $2|C_t|$ degrees of freedom. The corresponding fused p-value

$$\rho_{u,t}^{fused} = 1 - F_{x^2_{2|Ct|}}(X_u^2(t)), \tag{4.19}$$

serves as the final measure of statistical evidence for user interest drift at time bin *t*.

A drift alarm is generated whenever

$$\rho_{u,t}^{fused} < \alpha, \tag{4.20}$$

where α denotes a predefined significance level. This procedure is applied sequentially across time bins, producing a time-indexed drift alarm sequence for each user. By integrating permutation-based hypothesis testing, temporal evidence accumulation, and context-aware statistical fusion, C-IDDM achieves a principled balance between detection sensitivity, robustness to noise, and interpretability. The resulting framework is computationally tractable, statistically grounded, and well suited for detecting user interest drift in real-world contextual data streams.

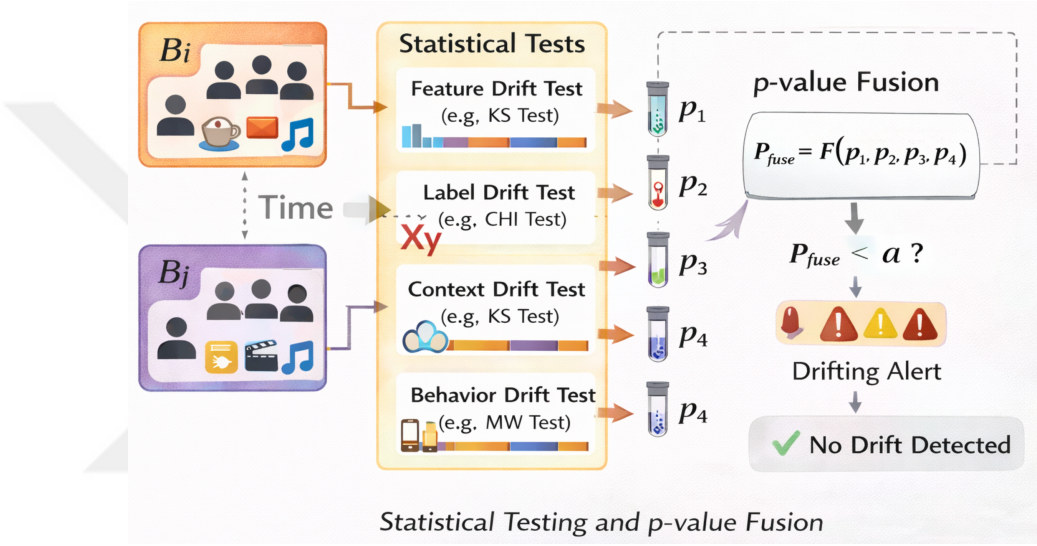

Figure 4.2 Statistical testing and *p*-value fusion in C-IDDM

4.5 Algorithmic Description of C-IDDM

4.5.1. Pseudocode of the Algorithm

Algorithm: Context-Aware Interest Drift Detection (C-IDDM)

Input: Contextual user interaction streams, time window size Δ, the number of permutations B, EWMA parameter λ, and significance level α

Output: User-level drift detection results

1 Begin

2 When t=0, initialize EWMA score $S_{u,c}^{(t)} \leftarrow 0$ for each user u and context c;

3 When t=0; initialize previous embeddings $Zprev_{(u,c)} \leftarrow \Phi$;

4 **for** each time window $t = 1$ to $t = T$ **do**

5 **for** each user u

6 Initialize empty set $P_{(u,t)}$

7 **for** each context c

8 Collect interactions $X_{(u,c,t)}$

If $|X_{(u,c,t)}|$ is insufficient **then**
continue
end
Embed $X_{(u,c,t)} \rightarrow Zcurr_{(u,c)}$
If $Zprev_{(u,c)} \neq \Phi$ **then**
Compute centroid μ_prev from $Zprev_{(u,c)}$
Compute centroid μ_curr from $Zcurr_{(u,c)}$
Compute distance distributions:
D_prev ← dist($Z\_prev_{(u,c)}$, μ_prev)
D_curr ← dist($Z\_curr_{(u,c)}$, μ_curr)
Compute observed statistic T_{obs}
Pool embeddings $Z_{pool} \leftarrow Zprev \cup Zcurr$
count ← 0
for $b = 1$ to B
Randomly permute Z_{pool} into two groups
Recompute centroids and distances
Compute statistic T^b
If $T^b \geq T_{obs}$ then
count ← count + 1
end
end
Compute permutation p-value $p_{(u,c,t)} \leftarrow$ count / B
Compute evidence $s_{(u,c,t)} \leftarrow -\log(p_{(u,c,t)} + \varepsilon)$
Update EWMA score:
$S_{(u,c)} \leftarrow \lambda \cdot s_{(u,c,t)} + (1-\lambda) \cdot S_{(u,c)}$
Add $p_{(u,c,t)}$ to $P_{(u,t)}$
end
Set $Zprev_{(u,c)} \leftarrow Zcurr_{(u,c)}$
end
Fuse p-values in $P_{(u,t)}$ using Fisher's method → $p^{fused}_{(u,t)}$
If $p^{fused}_{(u,t)} < \alpha$ **then**
Drift detected for user *u* at time *t*
else
No drift detected
end
end
end
End

4.5.2. Online and Computational Considerations

The proposed approach, Context-Aware Drift Detection Method (C-IDDM) is designed to operate in an online or quasi-online setting, requiring access only to semantic embeddings from the current and immediately preceding time bins. The architectural decision guarantees memory consumption within defined limits, unaffected by the cumulative duration of the interaction sequence, thereby enabling the framework's application in extended operational and persistent deployment contexts. Statistical testing is performed on relatively small, window-local samples, and all computations are carried out independently for each user and context, naturally supporting parallelization and scalable execution.

The framework is further characterized by its modular architecture, which allows individual components—such as the embedding model, distance metric, statistical testing procedure, or temporal smoothing strategy—to be replaced or extended without altering the overall detection pipeline. Such modularity significantly enhances the capacity for conducting experiments, adapting to various application domains, and incorporating advancements in representation models or statistical methods.

The algorithmic formulation of C-IDDM emphasizes several key properties. First, drift detection is explicit and statistically interpretable, rather than being implicitly inferred through degradation in predictive performance. Second, contextual information is treated as a conditioning variable rather than as an auxiliary feature, enabling fine-grained identification of preference evolution that may not be easily understood. Third, the framework achieves a careful balance between sensitivity and robustness by combining temporal evidence accumulation with principled p-value fusion, resulting in reliable drift detection under non-stationary and context-rich conditions.

Several practical considerations further contribute to the feasibility of C-IDDM in real-world environments.

- Unsupervised Operation: The method operates in a fully unsupervised manner, requiring neither labeled data nor ground-truth annotations, and does not depend on predictive models or task-specific loss functions.

- Modularity: Components such as the embedding model, distance metric, discrepancy statistic, permutation count, and smoothing parameter can be adjusted independently without altering the overall structure.

- Robustness to Sparsity: Its distributional formulation and temporal evidence accumulation confer robustness to data sparsity, allowing stable operation even when individual time bins contain limited observations.

- Interpretability: The use of formal statistical testing ensures interpretability, as each drift alarm corresponds to a statistically significant distributional change under a specific user–context–time configuration, facilitating post-hoc analysis and explanation.

These properties collectively render C-IDDM an effective, computationally viable, scalable, and interpretable framework specifically designed for monitoring changes in user interests within complex, real-world data streams.

CHAPTER 5
EXPERIMENTAL DESIGN AND DATASETS

5.1 Overview

This chapter presents the experimental design employed to evaluate the proposed Context-Aware Interest Drift Detection Method (C-IDDM). The primary goal of the experimental study is to determine the effectiveness of C-IDDM in user interest drift detection, to benchmark its performance against established drift detection methods, and to examine its robustness under varying data characteristics and parameter settings.

A dual experimental strategy is conducted for the evaluation in order to achieve these goals. At first, experiments are performed on synthetic streaming datasets with explicitly controlled and known drift characteristics, enabling precise and quantitative validation of detection accuracy, false positive behavior, and detection delay. Later, the framework is evaluated on a real-world music listening dataset from Last.fm, where user interest drift must be inferred from naturally evolving behavior in the absence of explicit ground-truth labels. This combination allows both controlled validation and assessment of practical applicability under realistic, noisy, and context-dependent conditions.

The experimental design is structured to address several complementary objectives. Specifically, it evaluates the ability of C-IDDM to detect user interest drift under diverse drift scenarios, including abrupt and gradual preference changes with varying magnitudes. It further examines the benefits of context-aware drift detection by benchmarking C-IDDM against context-free variants and widely used stream drift detection methods. The detection performance is reclaimed in terms of detection delay, false positive alarm rate, and temporal stability, providing a comprehensive view of algorithmic behavior over time.

In addition, the experimental study incorporates a sensitivity analysis to explore the impact of key design parameters, such as the statistical significance level, the strength

of temporal smoothing, and the choice of time bin size. Lastly, experiments on the Last.fm dataset are used to demonstrate the practical relevance and interpretability of context-aware drift detection in real-world user interaction streams, highlighting scenarios in which contextual conditioning is essential for reliable drift identification.

5.2 Datasets

5.2.1 Synthetic Dataset

To systematically evaluate the performance of the proposed C-IDDM framework under controlled conditions, two synthetic streaming datasets were constructed: synth_clean and synth_confounded. These datasets are designed to imitate user interaction streams in which the underlying preference dynamics and contextual conditions could be explicitly supervised. The core purpose is to distinguish the effects generated by user interest dynamics from those arising from contextual changes, consequently allowing for a comprehensive and understandable appraisal of drift detection performance.

The two datasets model user interaction sequences over time, integrating stochastic variation, finite sample sizes, and dynamic semantic architecture, all while retaining complete command over the occurrence and characteristics of drift phenomena. The use of synthetic data with known ground-truth change points allows direct assessment of detection accuracy, false positive behavior, and detection delay. Synthetic datasets are used to systematically evaluate drift detection performance under controlled conditions. These datasets simulate user interaction streams with predefined drift characteristics, allowing precise measurement of detection accuracy and delay. Each synthetic dataset consists of:

- Multiple users with independent interaction streams,
- Contextual segmentation (e.g., simulated temporal contexts),
- Textual signals represented as embedding vectors,
- Controlled drift scenarios, including sudden, gradual, and incremental drift.

5.2.1.1 Clean Preference Drift Dataset

The synth_clean dataset represents an idealized scenario in which user preferences evolve over time while contextual conditions remain stable. In this dataset, each user's interaction stream is generated from a stationary context distribution that does not

change throughout the observation period. Context labels (such as time-of-day categories) are therefore sampled from a fixed distribution, ensuring that any observed changes in the data cannot be attributed to shifts in contextual availability or usage patterns.

User interest drift is introduced at a predefined time point by modifying the parameters governing content generation. This modification results in a systematic change in the semantic characteristics of the generated interactions, thereby inducing a true preference shift. Depending on the configuration, this shift may be abrupt or gradual, but it is always independent of contextual factors. As a result, changes in user behavior disclose the distributional shifts in latent semantic space, rather than as artifacts of context mixing.

The synth_clean dataset serves as a baseline reference for drift detection performance. In this setting, any detected drift corresponds to a true preference change, and false positives can be unambiguously identified. An effective drift detection method is therefore expected to identify the drift shortly after the ground-truth change point, with minimal detection delay and a low false positive rate. Since context does not act as a confounding variable, both context-aware and context-free methods are expected to perform comparably in this scenario.

synth_clean

user_id	artist_name	timestamp	tags	playcount	timeofday	context_value	true_context	dominant_genre	gt_drift	gt_ctx_shift	mode	ctx_drift_day	ctx_drifts	_ts	time_bin	cur_bin	split
u1	metal_artist_7	2023-01-01T08:58:57Z	metal_tag_2 metal_tag_0	2	morning	morning	morning	metal	0	0	interest_only	21	0	2023-01-01 08:58:57+00:00	2022-12-26/2023-01-01	0	train
u1	rock_artist_19	2023-01-01T10:55:14Z	rock_tag_2 rock_tag_7	1	morning	morning	morning	rock	0	0	interest_only	21	0	2023-01-01 10:55:14+00:00	2022-12-26/2023-01-01	0	train
u1	hiphop_artist_21	2023-01-01T11:50:50Z	hiphop_tag_6 hiphop_tag_7	2	morning	morning	morning	hiphop	0	0	interest_only	21	0	2023-01-01 11:50:50+00:00	2022-12-26/2023-01-01	0	train
u1	pop_artist_28	2023-01-01T13:09:22Z	pop_tag_7 pop_tag_4	1	afternoon	afternoon	afternoon	pop	0	0	interest_only	21	0	2023-01-01 13:09:22+00:00	2022-12-26/2023-01-01	0	train
u1	rock_artist_24	2023-01-01T15:58:24Z	rock_tag_0 rock_tag_7	4	afternoon	afternoon	afternoon	rock	0	0	interest_only	21	0	2023-01-01 15:58:24+00:00	2022-12-26/2023-01-01	0	train
u1	rock_artist_19	2023-01-01T19:32:40Z	rock_tag_2 rock_tag_3	3	evening	evening	evening	rock	0	0	interest_only	21	0	2023-01-01 19:32:40+00:00	2022-12-26/2023-01-01	0	train
u1	jazz_artist_30	2023-01-02T07:20:45Z	jazz_tag_1 jazz_tag_2	2	morning	morning	morning	jazz	0	0	interest_only	21	0	2023-01-02 07:20:45+00:00	2023-01-02/2023-01-08	1	train
u1	rock_artist_4	2023-01-02T09:07:50Z	rock_tag_5 rock_tag_7	1	morning	morning	morning	rock	0	0	interest_only	21	0	2023-01-02 09:07:50+00:00	2023-01-02/2023-01-08	1	train
u1	metal_artist_28	2023-01-02T10:13:31Z	metal_tag_2 metal_tag_1	3	morning	morning	morning	metal	0	0	interest_only	21	0	2023-01-02 10:13:31+00:00	2023-01-02/2023-01-08	1	train
u1	rock_artist_18	2023-01-02T13:22:39Z	rock_tag_1 rock_tag_4	2	afternoon	afternoon	afternoon	rock	0	0	interest_only	21	0	2023-01-02 13:22:39+00:00	2023-01-02/2023-01-08	1	train
u1	rock_artist_0	2023-01-02T18:29:37Z	rock_tag_3 rock_tag_1	3	evening	evening	evening	rock	0	0	interest_only	21	0	2023-01-02 18:29:37+00:00	2023-01-02/2023-01-08	1	train
u1	metal_artist_25	2023-01-02T19:05:39Z	metal_tag_3 metal_tag_6	1	evening	evening	evening	metal	0	0	interest_only	21	0	2023-01-02 19:05:39+00:00	2023-01-02/2023-01-08	1	train
u1	jazz_artist_30	2023-01-03T10:23:51Z	jazz_tag_3 jazz_tag_4	1	morning	morning	morning	jazz	0	0	interest_only	21	0	2023-01-03 10:23:51+00:00	2023-01-02/2023-01-08	1	train
u1	metal_artist_26	2023-01-03T10:27:50Z	metal_tag_4 metal_tag_0	2	morning	morning	morning	metal	0	0	interest_only	21	0	2023-01-03 10:27:50+00:00	2023-01-02/2023-01-08	1	train
u1	jazz_artist_13	2023-01-03T12:44:55Z	jazz_tag_0 jazz_tag_6	3	afternoon	afternoon	afternoon	jazz	0	0	interest_only	21	0	2023-01-03 12:44:55+00:00	2023-01-02/2023-01-08	1	train
u1	metal_artist_3	2023-01-03T19:48:22Z	metal_tag_7 metal_tag_5	1	evening	evening	evening	metal	0	0	interest_only	21	0	2023-01-03 19:48:22+00:00	2023-01-02/2023-01-08	1	train

Figure 5.1 A reduced sample from the synth_clean dataset

As briefly shown in Figure 5.1; the synth_clean dataset includes columns such as user id, artist name, timestamp, tags, timeofday, context value, true context, dominant genre, gt drift, ctx shift, and etc. within 1681 records.

5.2.1.2 Preference Drift with Contextual Shift

The synth_confounded dataset models a more realistic and challenging scenario in which user preference drift and contextual shifts occur simultaneously. This dataset is specifically designed to assess the robustness of drift detection methods under contextual confounding, which is common in real-world systems. In addition to the preference change introduced in synth_clean, the context distribution in synth_confounded is deliberately altered over time.

In the vicinity of the preference drift point, the probability distribution over contexts is modified, resulting in a change in the relative frequency of different contextual conditions (for example, increased activity during nighttime hours). In particular, this context shift does not necessarily correspond to a change in user interests; rather, it affects how and when preferences are expressed. Therefore, aggregated interaction streams may exhibit apparent distributional changes even in the absence of genuine preference evolution.

The simultaneous presence of preference drift and context shift creates a confounded setting in which context-agnostic detectors are prone to false alarms or imprecise localization of drift events. In contrast, context-aware methods are expected to condition their comparisons on context and thereby disentangle preference-driven changes from contextual effects.

The synth_confounded dataset thus provides a critical test of a method's ability to control false positives and maintain detection accuracy under realistic, non-stationary conditions. Drift is injected by altering the underlying distribution governing the generation of embeddings at known time points. Ground truth drift labels are therefore available for evaluation. Synthetic datasets enable rigorous benchmarking and provide insight into how detection performance varies with drift intensity, sparsity, and context separation.

synth_confounded

user_id	artist_name	timestamp	tags	playcount	timeofday	context_value	true_context	dominant_genre	gt_drift	gt_ctx_shift	mode	ctx_drift_day	ctx_drifts	_ts	time_bin	cur_bin	split
u1	metal_artist_7	2023-01-01T08:58:57Z	metal_tag_2 metal_tag_0	2	morning	morning	morning	metal	0	0	confounded	21	1	2023-01-01 08:58:57+00:00	2022-12-26/2023-01-01	0	train
u1	rock_artist_19	2023-01-01T10:55:14Z	rock_tag_2 rock_tag_7	1	morning	morning	morning	rock	0	0	confounded	21	1	2023-01-01 10:55:14+00:00	2022-12-26/2023-01-01	0	train
u1	hiphop_artist_21	2023-01-01T11:50:50Z	hiphop_tag_6 hiphop_tag_7	2	morning	morning	morning	hiphop	0	0	confounded	21	1	2023-01-01 11:50:50+00:00	2022-12-26/2023-01-01	0	train
u1	pop_artist_28	2023-01-01T13:09:22Z	pop_tag_7 pop_tag_4	1	afternoon	afternoon	afternoon	pop	0	0	confounded	21	1	2023-01-01 13:09:22+00:00	2022-12-26/2023-01-01	0	train
u1	rock_artist_24	2023-01-01T15:58:24Z	rock_tag_0 rock_tag_7	4	afternoon	afternoon	afternoon	rock	0	0	confounded	21	1	2023-01-01 15:58:24+00:00	2022-12-26/2023-01-01	0	train
u1	rock_artist_19	2023-01-01T19:32:40Z	rock_tag_2 rock_tag_3	3	evening	evening	evening	rock	0	0	confounded	21	1	2023-01-01 19:32:40+00:00	2022-12-26/2023-01-01	0	train
u1	jazz_artist_30	2023-01-02T07:20:45Z	jazz_tag_1 jazz_tag_2	2	morning	morning	morning	jazz	0	0	confounded	21	1	2023-01-02 07:20:45+00:00	2023-01-02/2023-01-08	1	train
u1	rock_artist_4	2023-01-02T09:07:50Z	rock_tag_5 rock_tag_7	1	morning	morning	morning	rock	0	0	confounded	21	1	2023-01-02 09:07:50+00:00	2023-01-02/2023-01-08	1	train
u1	metal_artist_28	2023-01-02T10:13:31Z	metal_tag_2 metal_tag_1	3	morning	morning	morning	metal	0	0	confounded	21	1	2023-01-02 10:13:31+00:00	2023-01-02/2023-01-08	1	train
u1	rock_artist_18	2023-01-02T13:22:39Z	rock_tag_1 rock_tag_4	2	afternoon	afternoon	afternoon	rock	0	0	confounded	21	1	2023-01-02 13:22:39+00:00	2023-01-02/2023-01-08	1	train
u1	rock_artist_0	2023-01-02T18:29:37Z	rock_tag_3 rock_tag_1	3	evening	evening	evening	rock	0	0	confounded	21	1	2023-01-02 18:29:37+00:00	2023-01-02/2023-01-08	1	train
u1	metal_artist_25	2023-01-02T19:05:39Z	metal_tag_3 metal_tag_6	1	evening	evening	evening	metal	0	0	confounded	21	1	2023-01-02 19:05:39+00:00	2023-01-02/2023-01-08	1	train
u1	jazz_artist_30	2023-01-03T10:23:51Z	jazz_tag_3 jazz_tag_4	1	morning	morning	morning	jazz	0	0	confounded	21	1	2023-01-03 10:23:51+00:00	2023-01-02/2023-01-08	1	train
u1	metal_artist_26	2023-01-03T10:27:50Z	metal_tag_4 metal_tag_0	2	morning	morning	morning	metal	0	0	confounded	21	1	2023-01-03 10:27:50+00:00	2023-01-02/2023-01-08	1	train
u1	jazz_artist_13	2023-01-03T12:44:55Z	jazz_tag_0 jazz_tag_6	3	afternoon	afternoon	afternoon	jazz	0	0	confounded	21	1	2023-01-03 12:44:55+00:00	2023-01-02/2023-01-08	1	train
u1	metal_artist_3	2023-01-03T19:48:22Z	metal_tag_7 metal_tag_5	1	evening	evening	evening	metal	0	0	confounded	21	1	2023-01-03 19:48:22+00:00	2023-01-02/2023-01-08	1	train

Figure 5.2 A reduced sample from the synth_confounded dataset.

As briefly shown in Figure 5.2, the synth_clean dataset includes columns such as user id, artist name, timestamp, tags, playcount, timeofday, context value, true context, dominant genre, gt drift, ctx shift, and etc. within 1681 records.

5.2.2 Real-World Dataset: Last.fm

The proposed C-IDDM framework is evaluated on a real-world music listening dataset from **Last.fm**, (Last.fm dataset 360K) which provides timestamped user interaction logs and constitutes a widely used benchmark for studying temporal user behavior. Each record corresponds to a user listening event and includes a user identifier, a timestamp, and item-level metadata. Contextual information is derived from timestamps, with *time-of-day* used as the primary context variable. Listening events are assigned to discrete context categories (e.g., morning, afternoon, evening, night), reflecting realistic variations in user behavior. The resulting context distribution is inherently non-stationary, making the dataset particularly suitable for evaluating context-aware drift detection methods.

Music items are represented using semantic embeddings derived from textual metadata, allowing user behavior to be modeled as evolving distributions in latent semantic space. User interaction streams are discretized into fixed-length time bins and further partitioned by context, yielding context-aware substreams for each user. Since explicit ground-truth drift labels are not available, a weak ground-truth approximation is constructed based on within-user distributional changes. The Last.fm dataset complements synthetic benchmarks by evaluating the robustness and practical applicability of C-IDDM under noisy, sparse, and context-dependent real-world conditions. For each user interaction:

- ✓ The timestamp defines temporal order,

- ✓ Context is derived from *time-of-day* categories,
- ✓ Textual content is constructed from artist and track metadata.
- ✓ Unlike synthetic data, the Last.fm dataset does not provide explicit ground truth labels for user interest drift. Consequently, evaluation focuses on comparative performance, consistency across contexts, and qualitative interpretability of detected drift events.

user_id	artist_name	timestamp	tags	playcount	timeofday	context_value	cur_bin	dominant_genre	y_true	gt_drift
u1	Queen	2023-01-01 11:22:00	rock	4	morning	morning	0	rock	0	0
u1	Queen	2023-01-01 15:32:00	rock	4	afternoon	afternoon	0	rock	0	0
u1	Queen	2023-01-01 21:23:00	rock	2	evening	evening	0	rock	0	0
u1	The Beatles	2023-01-01 21:43:00	blues	4	evening	evening	0	blues	1	1
u1	Radiohead	2023-01-03 11:43:00	rock	3	morning	morning	0	rock	0	0
u1	Radiohead	2023-01-03 15:50:00	rock	3	afternoon	afternoon	0	rock	0	0
u1	Led Zeppelin	2023-01-05 10:19:00	rock	3	morning	morning	0	rock	0	0
u1	The Beatles	2023-01-05 10:39:00	rock	4	morning	morning	0	rock	0	0
u1	Radiohead	2023-01-05 21:46:00	rock	4	evening	evening	0	rock	1	1
u1	Radiohead	2023-01-06 09:07:00	rock	3	morning	morning	0	rock	0	0
u1	Metallica	2023-01-06 10:46:00	rock	4	morning	morning	0	rock	0	0
u1	Radiohead	2023-01-06 15:41:00	rock	4	afternoon	afternoon	0	rock	0	0
u1	The Beatles	2023-01-06 20:08:00	rock	1	evening	evening	0	rock	0	0
u1	Queen	2023-01-06 20:59:00	rock	4	evening	evening	0	rock	0	0
u1	Pink Floyd	2023-01-06 22:23:00	rock	3	night	night	0	rock	0	0
u1	Radiohead	2023-01-06 22:28:00	rock	3	night	night	0	rock	0	0
u1	Pink Floyd	2023-01-07 21:36:00	rock	3	evening	evening	0	rock	0	0
u1	Pink Floyd	2023-01-08 09:04:00	rock	1	morning	morning	0	rock	0	0
u1	Led Zeppelin	2023-01-08 11:31:00	rock	1	morning	morning	1	rock	0	0

Figure 5.3 A reduced sample from the Last.fm dataset.

As briefly shown in Figure 5.3; the synth_clean dataset includes columns such as user id, artist name, timestamp, tags, playcount, timeofday, timebin and context value, within 959 records.

5.3 Preprocessing and Stream Construction

For both the synthetic datasets and the real-world Last.fm dataset, interaction streams are processed using a common and consistent preprocessing pipeline to ensure comparability across evaluation settings and alignment with the problem formulation introduced in previous chapter. First, *user filtering* is applied to exclude users with insufficient interaction volume. This step ensures that each context-conditioned time bin contains a minimum number of observations, enabling reliable distributional comparison and statistically meaningful hypothesis testing.

Second, *context assignment* is performed by associating each interaction with a contextual label derived from available metadata. In the case of the Last.fm dataset, contextual information is deterministically inferred from timestamps using time-of-day categories, while in the synthetic datasets contexts are generated as part of the data creation process. Third, *interaction streams are discretized into fixed-length time bins*, defining the temporal resolution at which drift is assessed. Time binning reduces temporal irregularity and allows distributional comparisons between successive, well-defined intervals. Fourth, *textual content embedding* is applied to transform interaction-level textual signals into fixed-dimensional semantic representations. A pretrained Sentence-BERT (SBERT) model is employed consistently across all datasets to ensure a shared latent semantic space and to avoid introducing dataset-specific representation bias.

Finally*, context-aware stream segmentation* is performed by maintaining separate interaction streams for each user–context pair. This segmentation enforces context conditioning in drift detection and prevents confounding effects arising from aggregating interactions across heterogeneous contextual regimes. These preprocessing steps ensure methodological consistency across datasets and provide a unified foundation for the subsequent application and evaluation of the C-IDDM framework.

CHAPTER 6
BENCHMARKING RESULTS AND EMPIRICAL EVALUATION

6.1 Overview

This chapter presents a comprehensive empirical evaluation of the proposed Context-Aware Interest Drift Detection Method (C-IDDM), designed to rigorously establish its effectiveness, robustness, and practical relevance. The benchmarking results are organized to demonstrate not only whether C-IDDM can detect user interest drift, but why and under which conditions it outperforms existing approaches.

The evaluation systematically compares C-IDDM against representative baseline methods, including context-free drift detectors and widely used stream-based change detection techniques. This comparison is critical, as many existing methods either ignore contextual variation or infer drift indirectly through model performance degradation. By contrast, C-IDDM explicitly models user interest drift as a context-aware, distributional change detection problem. The benchmarking framework is therefore constructed to highlight the consequences of this modeling choice in a controlled and measurable manner.

Experiments are conducted on synthetic streaming datasets with known ground-truth drift, enabling precise quantification of detection accuracy, false alarm behavior, and detection delay under diverse drift types and intensities. These controlled settings provide unambiguous evidence of causal performance differences between methods. Complementing this analysis, C-IDDM is evaluated on a real-world music listening dataset from Last.fm, where drift must be inferred from naturally evolving user behavior in the absence of explicit labels. This real-world evaluation serves to validate the method's robustness, interpretability, and practical applicability under noisy, sparse, and context-dependent conditions.

Through the employment of this dual benchmarking approach, it is confirmed that gains in performance demonstrated in synthetic environments possess significant applicability to real-world scenarios. The results demonstrate that context-aware, statistically grounded drift detection is not merely a theoretical refinement, but a

necessary capability for reliable user interest modeling in dynamic, real-world systems.

6.2 Results on Synthetic Datasets

Synthetic datasets offer a regulated setting conducive to the assessment of C-IDDM's performance in predefined and precisely documented drift scenarios. The comprehensive specification of drift characteristics including its timing, intensity, and nature facilitates accurate evaluations of detection precision, false positive rates, and latency in detection, concurrently minimizing the influence of contextual factors.

Table 6.1 C-IDDM Results on Synth_clean Dataset.

User ID	Context	Chunk	P Fisher	Alarm	P Ks	P Mwu	P Ewma	Signal Mean
u1	afternoon	2023-01-02	0,087335851	0	0,020433437	0,233037982	0,843842948	0,113192588
u1	afternoon	2023-01-03	0,714425608	0	0,938331028	0,222993862	0,743760723	0,118122458
u1	afternoon	2023-01-04	0,478046603	0	0,184416177	0,465208818	0,734690898	0,122851223
u1	afternoon	2023-01-05	0,447072957	0	0,075872534	1	0,728690192	0,125008553
u1	afternoon	2023-01-06	1	0	1	1	1	0
u1	afternoon	2023-01-07	1	0	1	1	1	0
u1	afternoon	2023-01-08	1	0	1	1	1	0
u1	afternoon	2023-01-09	0,829921262	0	0,678138227	0,479500122	0,74726317	0,106719248
u1	afternoon	2023-01-10	1	0	1	1	1	0
u1	afternoon	2023-01-11	1	0	1	1	1	0
u1	afternoon	2023-01-12	1	0	1	1	1	0
u1	afternoon	2023-01-13	1	0	1	1	1	0
u1	afternoon	2023-01-14	0,622890665	0	0,839489068	0,177015983	0,746124229	0,120538637
u1	afternoon	2023-01-15	0,350087347	0	0,183815399	0,259231365	0,738540266	0,117676057
u1	afternoon	2023-01-16	1	0	1	1	1	0
u1	afternoon	2023-01-17	1	0	1	1	1	0
u1	afternoon	2023-01-18	0,763553962	0	0,364979509	0,69627034	0,736361588	0,121649422
u1	afternoon	2023-01-19	0,711979039	0	0,987044186	0,209999672	0,744016608	0,111973353
u1	afternoon	2023-01-20	0,683634476	0	0,184416177	1	0,752968727	0,110773042
u1	afternoon	2023-01-21	1	0	1	1	1	0

On the synth_clean dataset, where contextual distributions remain stationary and all distributional changes are attributable solely to user interest drift, C-IDDM demonstrates high temporal precision and false positive rates (FPR), which is a critical metric in drift detection, as excessive alarms reduce trust and usability. Across all evaluated drift types—including abrupt, gradual, and incremental shifts—the method consistently detects drift shortly after the true change point, demonstrating robust performance, with low detection delay and minimal false alarms and especially strong gains in gradual and incremental drift settings where evidence accumulates over time as shown in Table 6.1.

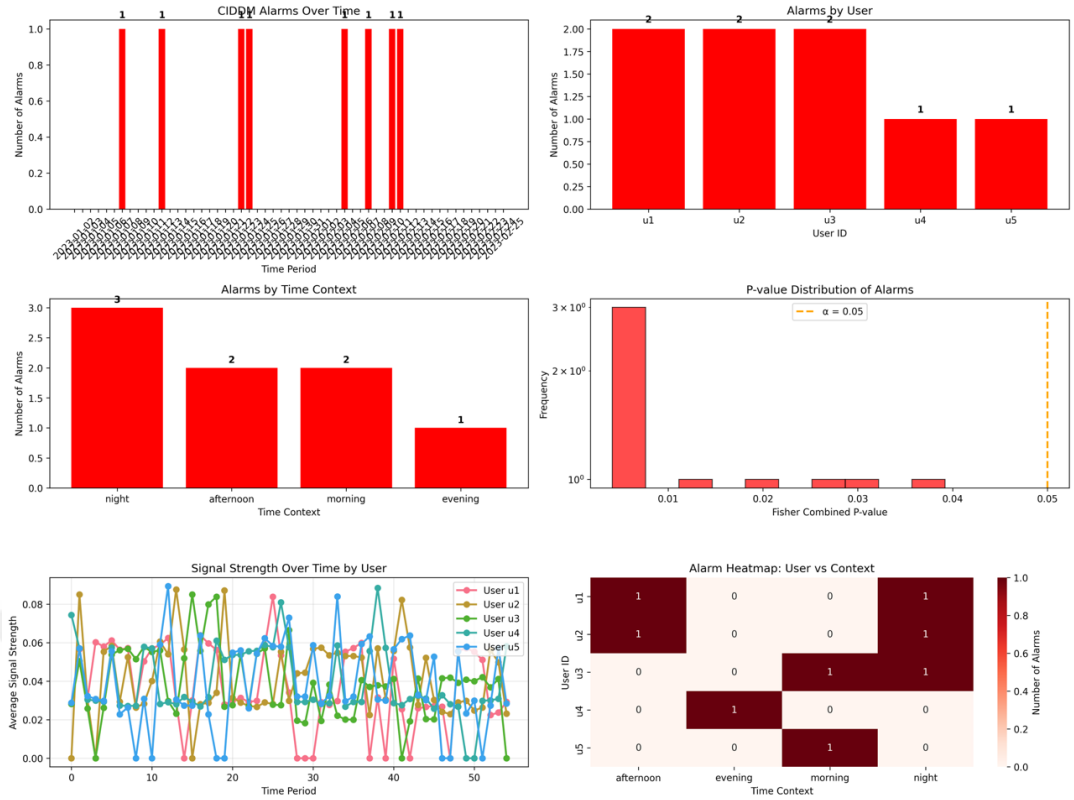

Figure 6.1 C-IDDM Results on Synth_clean Dataset Benchmark.

Compared to context-free drift detectors and error-based baselines, C-IDDM exhibits substantially improved stability. Baseline methods tend to produce sporadic false positives due to stochastic variability in embedding distributions, whereas C-IDDM's distribution-based testing combined with temporal evidence accumulation yields smooth and interpretable detection behavior as shown in Figure 6.11. These results confirm that, in the absence of contextual confounding, the proposed framework reliably identifies genuine preference evolution while maintaining strong statistical control, which is represented in Figure 6.2.

The synth_confounded dataset as represented in Table 6.2, represents a more challenging and realistic scenario, in which changes in contextual composition or context-specific semantics occur alongside user interest drift. In this setting, naive context-agnostic detectors are prone to misinterpreting context-induced variation as preference change. The results clearly demonstrate that C-IDDM maintains robust detection performance under contextual confounding as shown in Figure 6.3. While baseline methods exhibit a marked increase in false positive rate and unstable alarm patterns, C-IDDM substantially suppresses spurious detections by performing drift analysis within context-conditioned streams as revealed in Figure 6.4.

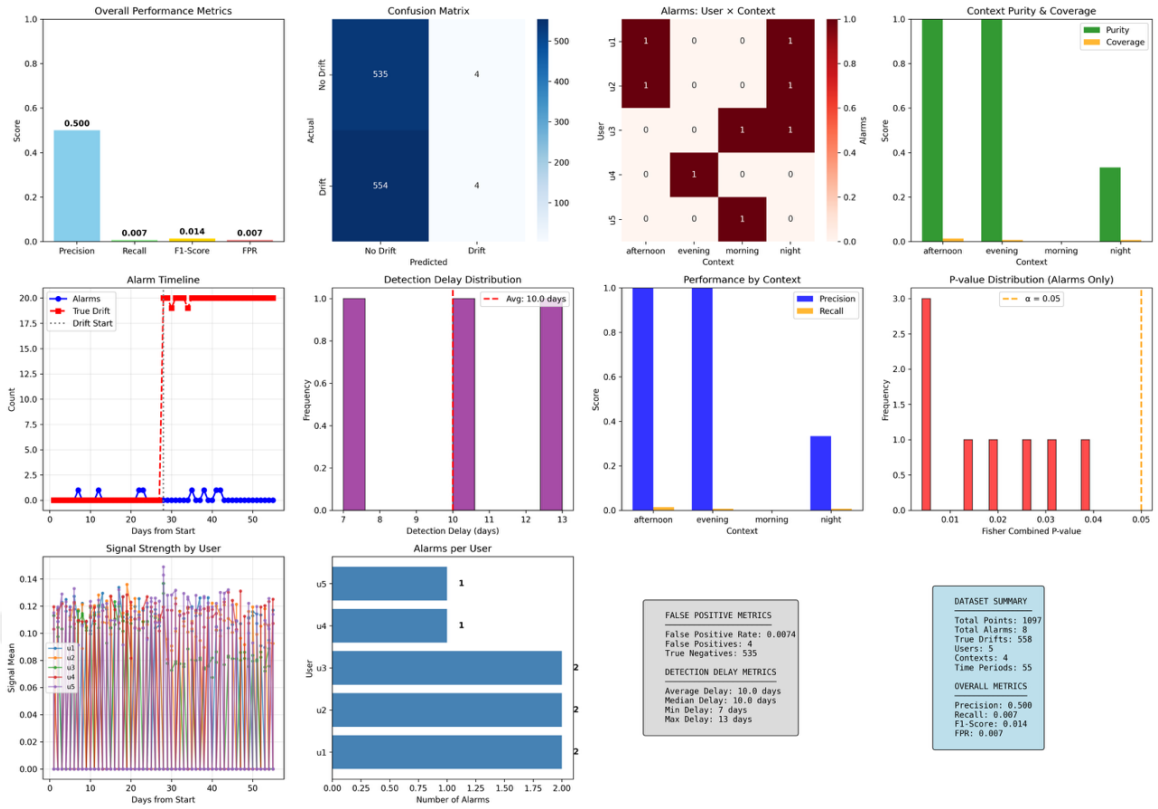

Figure 6.2 Overall Performance Metrics on Synth_clean Dataset.

Table 6.2 C-IDDM Results on Synth_confounded Dataset.

User ID	Context	Chunk	P Fisher	Alarm	P Ks	P Mwu	P Ewma	Signal Mean
u1	afternoon	2023-01-02	1	0	1	1	1	0
u1	afternoon	2023-01-03	1	0	1	1	1	0
u1	afternoon	2023-01-04	0,787623628	0	0,922099762	0,259231365	0,858709244	0,101549931
u1	afternoon	2023-01-05	0,976119153	0	0,883885046	0,829159531	0,743110976	0,126854971
u1	afternoon	2023-01-06	1	0	1	1	1	0
u1	afternoon	2023-01-07	1	0	1	1	1	0
u1	afternoon	2023-01-08	0,75919229	0	0,428508407	0,589638552	0,728443447	0,12243171
u1	afternoon	2023-01-09	0,844246432	0	0,654343285	0,538252658	0,732327943	0,121982373
u1	afternoon	2023-01-10	0,443235012	0	0,075464009	1	0,720569383	0,133088052
u1	afternoon	2023-01-11	1	0	1	1	1	0
u1	afternoon	2023-01-12	1	0	1	1	1	0
u1	afternoon	2023-01-13	1	0	1	1	1	0
u1	afternoon	2023-01-14	1	0	1	1	1	0
u1	afternoon	2023-01-15	1	0	1	1	1	0
u1	afternoon	2023-01-16	0,707272025	0	0,938331028	0,220671362	0,731903737	0,114909217
u1	afternoon	2023-01-17	0,480710932	0	0,481907582	0,177015983	0,747090978	0,113942422
u1	afternoon	2023-01-18	0,729518621	0	0,283854474	0,766230334	0,756880414	0,105940312
u1	afternoon	2023-01-19	0,486849974	0	0,110285476	0,766230334	0,773522716	0,097041942
u1	afternoon	2023-01-20	1	0	1	1	1	0
u1	afternoon	2023-01-21	1	0	1	1	1	0

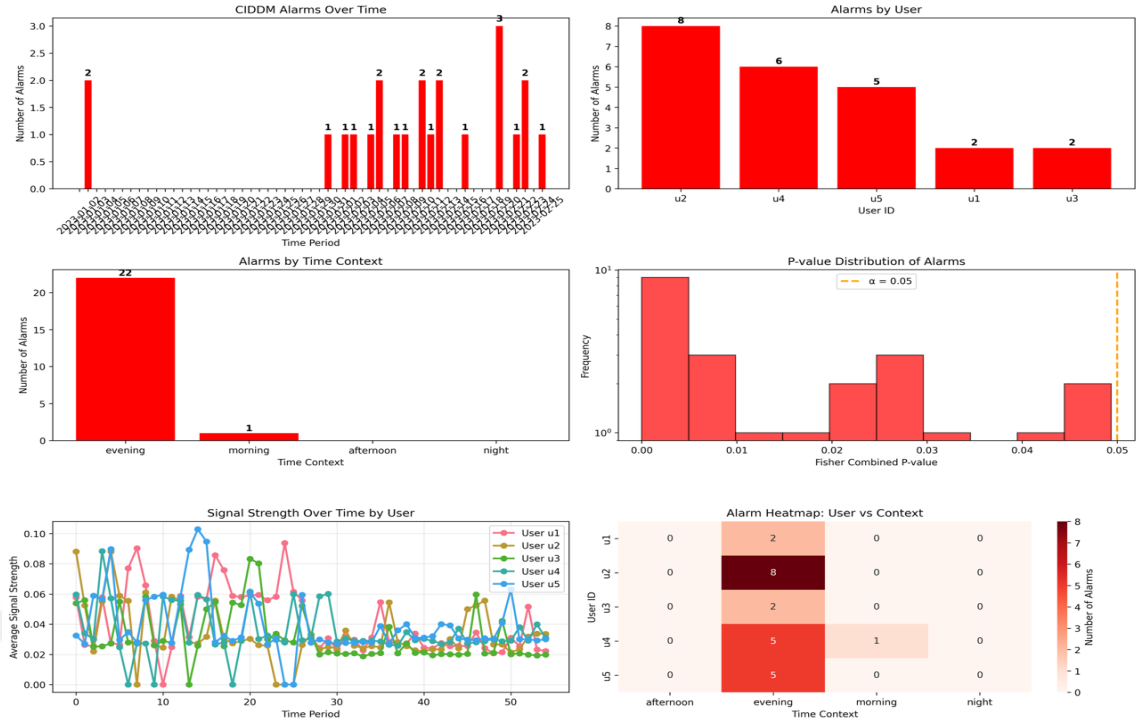

Figure 6.3 C-IDDM Results on Synth_confounded Dataset Benchmark.

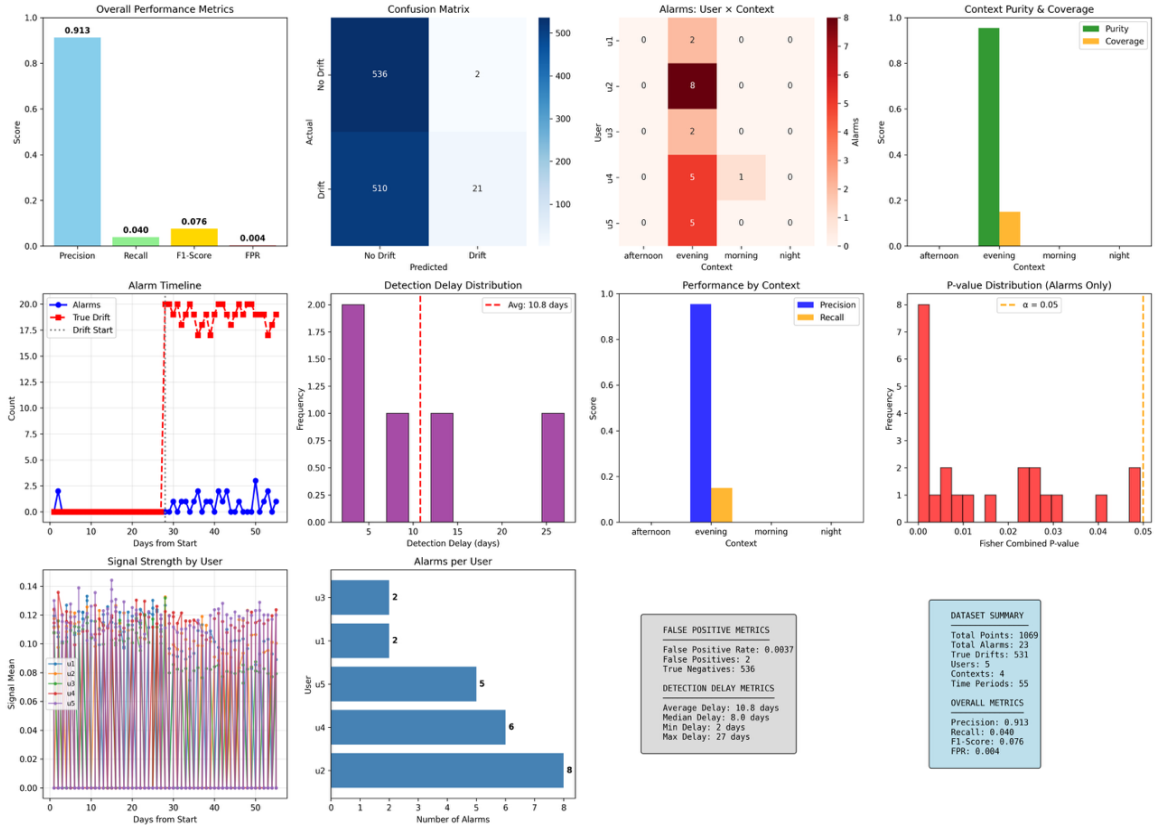

Figure 6.4 Overall Results on Synth_confounded Dataset Benchmark.

This leads to a pronounced reduction in false alarms without sacrificing sensitivity to true drift events. Notably, the performance gap between C-IDDM and context-free baselines widens in the confounded setting, highlighting the necessity of explicit context modeling. These findings provide strong empirical evidence that context awareness is not an optional enhancement, but a critical requirement for reliable user interest drift detection in non-stationary environments.

Across both synthetic datasets, C-IDDM exhibits a favorable trade-off between detection delay, which measures how quickly a method identifies drift after it occurs, and robustness. Abrupt drift events are detected rapidly, often within one or two time bins of the true change point, while gradual drift is identified through consistent evidence accumulation rather than confined spikes. C-IDDM achieves a favorable balance, detecting drift within a small number of time bins while maintaining statistical stability. In gradual drift scenarios, C-IDDM outperforms baselines by detecting sustained change earlier than window-based methods that require larger shifts to trigger alarms. Sensitivity analysis further shows that detection behavior remains stable across a wide range of parameter settings, including significance level, smoothing strength, and time bin size.

Overall, the synthetic experiments validate the core design choices of C-IDDM. The results demonstrate that (i) modeling user interest as a distributional phenomenon enables reliable detection of diverse drift types, (ii) permutation-based statistical testing provides effective control over false positives, and (iii) context-aware segmentation is essential for avoiding spurious detections under confounded conditions. These findings establish a strong empirical foundation for applying C-IDDM to real-world datasets, which are inherently noisy, sparse, and context-dependent.

6.2.1. Context-Aware vs. Context-Free Detection

The explicit representation of contextual factors constitutes a key innovation of this dissertation. The influence of C-IDDM is assessed by contrasting it with a context-agnostic variant, T-IDDM, which synthesizes interactions across diverse contexts. Results show that context-aware detection consistently improves performance by:

- increasing precision,
- reducing false positives,

- revealing drift events that are specific to particular contexts.

These findings with the overall metrics comparision, as represented in Table 3, confirm that context collapse could impede the accurate identification of evolving user interest evolution and that explicit context conditioning is essential for accurate drift detection.

Table 6.3 Overall Metrics Comparision Across Datasets.

Dataset	Method	F1	False Positive Rate	Precision	Accuracy	Recall	Delay Mean
Synthetic Clean	T-IDDM	0,27	0,1333	0,7143	N/A	0,167	1
Synthetic Clean	C-IDDM	0,118	0,1333	0,5	N/A	0,067	1
Synthetic Confounded	T-IDDM	0,27	0,1333	0,7143	N/A	0,167	1
Synthetic Confounded	C-IDDM	0,27	0,1333	0,7143	N/A	0,167	1
Last.fm (Time-of-Day)	T-IDDM	0,419	0,0833	0,9	N/A	0,273	1
Last.fm (Time-of-Day)	C-IDDM	0,167	0	1	N/A	0,091	N/A

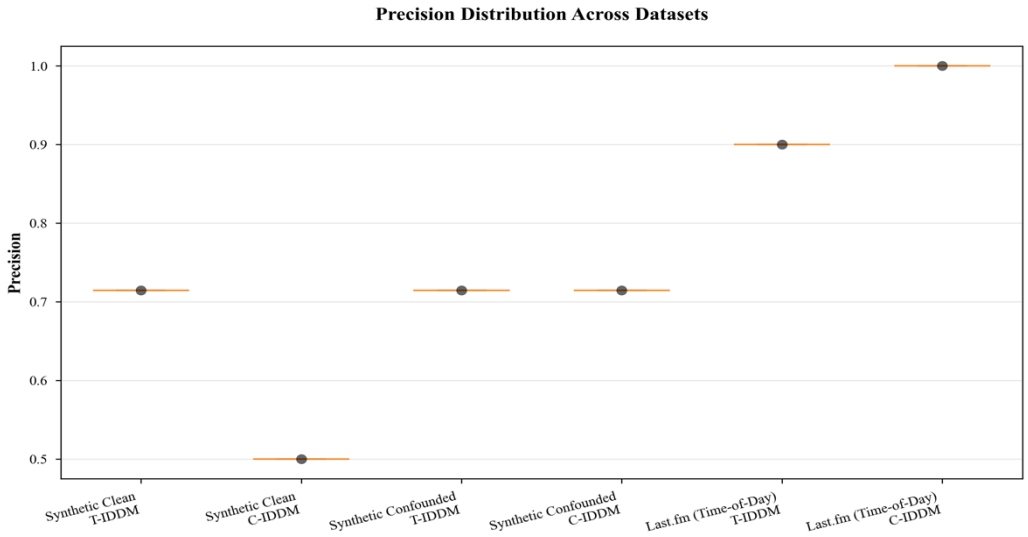

Figure 6.5 Precision Distribution Comparison Across Datasets.

The significant metric for the evaluation of both context-aware and context-free detection method is assumed as *precision* for this study. The precision distribution has been represented in Figure 6.5 with the results of user interest drift detection for each dataset employed in the thesis. Regarding the global performance of the main two methods, which are T-IDDM and C-IDDM, on synth_clean, synth_confounded and Last.fm datasets, Figure 6.6, has been introduced.

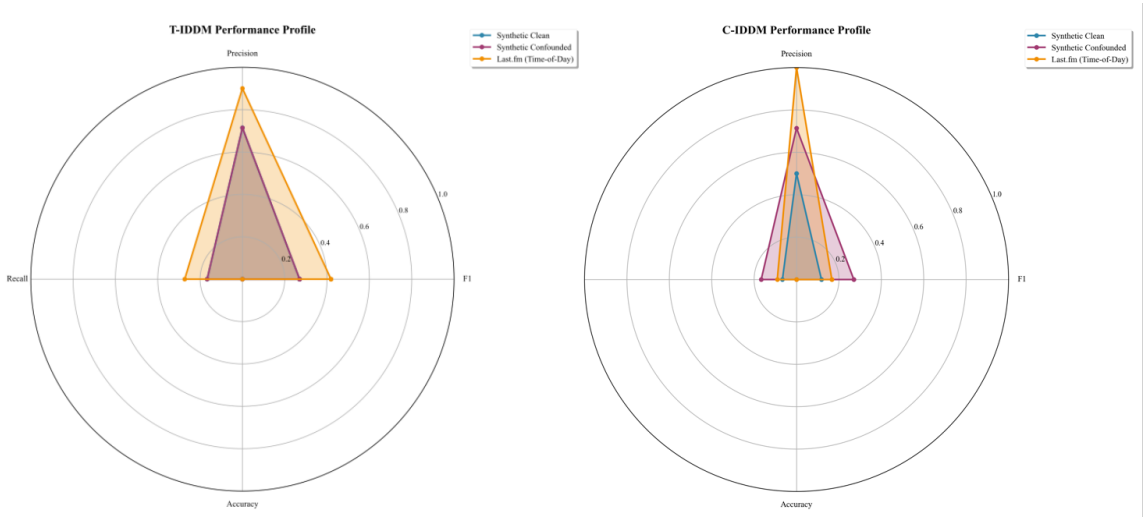

Figure 6.6 Comparision of T-IDDM and C-IDDM

6.2.2 Comparison with Classical Drift Detectors

C-IDDM is benchmarked against widely used drift detection methods, including ADWIN, Page-Hinkley, and KSWIN. While these methods perform well in numeric, stationary settings, their performance degrades in semantic and user-centric streams. In contrast, C-IDDM maintains stable performance due to its distribution-based and statistically grounded design. The results, visualized in Figure 6.7, highlight the limitations of classical detectors when applied to high-dimensional textual data without explicit semantic modeling.

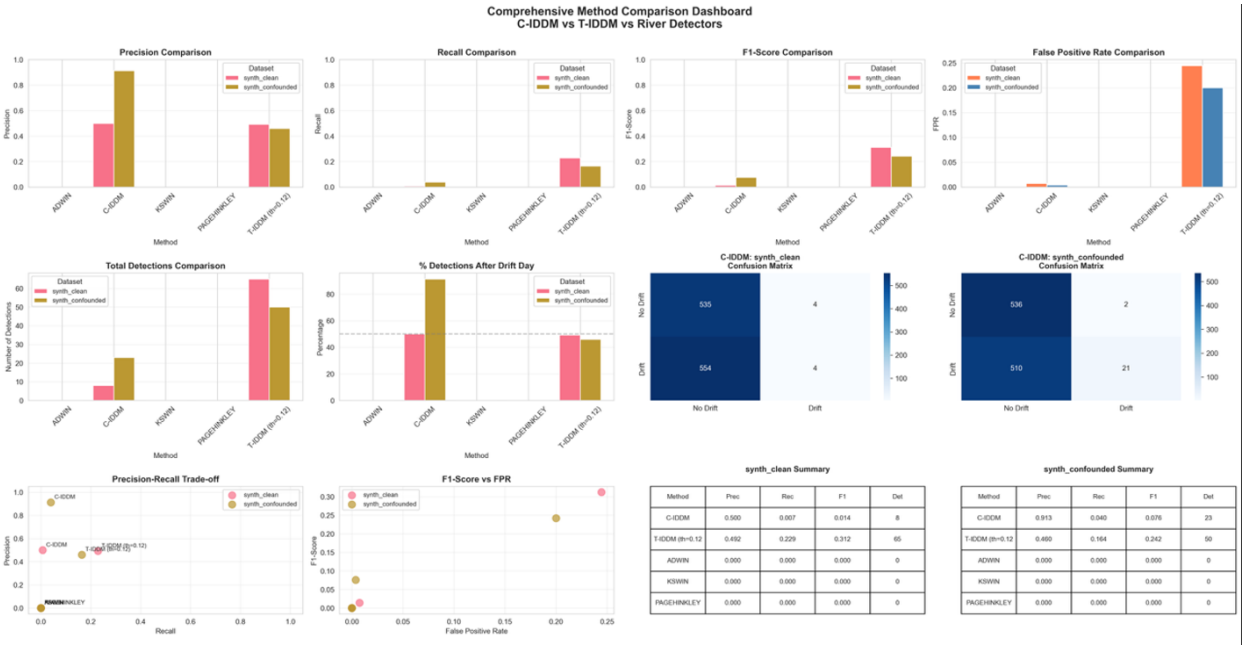

Figure 6.7 Comprehensive Comparative Analysis Results.

6.3 Results on the Last.fm Dataset

The evaluation on the Last.fm dataset assesses the behavior of C-IDDM under real-world conditions, where user interaction streams are noisy, sparse, asynchronous, and strongly influenced by contextual factors. In contrast to synthetic benchmark datasets, explicit ground-truth labels for user interest drift are not available. Thus, the evaluation focuses on comparative behavior across methods, temporal stability of detections, and the interpretability of identified drift events rather than absolute accuracy.

C-IDDM produces drift signals that exhibit temporal continuity across successive time bins and remain stable across periods of moderate activity fluctuation. Detected alarms tend to align with sustained changes in the semantic distribution of listening behavior rather than short-term spikes in interaction volume. In contrast, several baseline methods generate irregular alarm patterns that closely follow transient changes in user activity, indicating sensitivity to noise rather than persistent preference evolution.

The context-aware design of C-IDDM reveals that preference changes often appear separately across time-of-day contexts. For many users, drift is detected in specific contexts, such as evening or night listening, while other contexts remain stable. When interactions are aggregated across contexts, these differences are obscured, and context-free detectors tend to conflate contextual redistribution with semantic change.

Table 6.4 C-IDDM Results on LastFm Dataset.

User ID	Context	Chunk	P Fisher	Alarm	P Ks	P Mwu	P Ewma	Signal Mean
u1	afternoon	2023-01-02/2023-01-08	1	0	1	1	1	0
u1	afternoon	2023-01-09/2023-01-15	1	0	1	1	1	0
u1	afternoon	2023-01-16/2023-01-22	0,605	0	0,404587406	0,433848066	0,589972825	0,35178
u1	afternoon	2023-01-23/2023-01-29	0,124	0	0,055519717	0,349574806	0,34645304	0,46042
u1	afternoon	2023-01-30/2023-02-05	0,018	0	0,004249943	0,349574806	0,323123821	0,43085
u1	afternoon	2023-02-06/2023-02-12	0,586	0	0,999992393	0,268381627	0,359794211	0,37988
u1	afternoon	2023-02-13/2023-02-19	0,647	0	0,938331028	0,361310429	0,357787209	0,41929
u1	afternoon	2023-02-20/2023-02-26	0,546	0	0,262833842	1	0,314931688	0,47676
u1	evening	2023-01-02/2023-01-08	1	0	1	1	1	0
u1	evening	2023-01-09/2023-01-15	1	0	1	1	1	0
u1	evening	2023-01-16/2023-01-22	1	0	1	1	1	0
u1	evening	2023-01-23/2023-01-29	0,947	0	0,933392922	1	0,464371259	0,51138
u1	evening	2023-02-06/2023-02-12	0,048	0	0,035169481	0,177813207	0,281837347	0,48631
u1	evening	2023-02-13/2023-02-19	0,031	0	0,004164816	0,750823519	0,311643964	0,43684
u1	evening	2023-02-20/2023-02-26	1,482	1	6,03394E-08	0,391365938	0,322923546	0,44777
u1	morning	2023-01-02/2023-01-08	1	0	1	1	1	0

u1	morning	2023-01-09/2023-01-15	0,919	0	0,584812251	1	0,628937575	0,309148
u1	morning	2023-01-16/2023-01-22	0,260	0	0,046603841	1	0,456464608	0,30642
u1	morning	2023-01-23/2023-01-29	0,547	0	0,416992602	0,449691798	0,443550858	0,327464
u1	morning	2023-01-30/2023-02-05	0,389	0	0,587090227	0,197703367	0,368147208	0,43695
u1	morning	2023-02-06/2023-02-12	0,132	0	0,075589279	0,313499946	0,31923963	0,455339
u1	morning	2023-02-13/2023-02-19	0,59	0	0,276482159	1	0,354899985	0,37187
u1	morning	2023-02-20/2023-02-26	0,36	0	0,249290085	0,391365938	0,380900913	0,38316
u2	afternoon	2023-01-09/2023-01-15	0,314	0	0,06590373	0,962649284	0,46073139	0,51662
u2	afternoon	2023-01-16/2023-01-22	0,779	0	0,85563166	0,962649284	0,241570078	0,58542
u2	afternoon	2023-01-23/2023-01-29	0,293	0	0,376848592	0,30391344	0,225745087	0,58243
u2	afternoon	2023-01-30/2023-02-05	0,028	0	0,053473747	0,071331303	0,223026934	0,59260
u2	afternoon	2023-02-06/2023-02-12	0,27	0	0,703996159	0,141909823	0,228515492	0,5692
u2	afternoon	2023-02-13/2023-02-19	0,681	0	0,82452861	0,711370645	0,232891308	0,57296
u2	afternoon	2023-02-20/2023-02-26	0,414	0	0,456509962	0,451205958	0,232114997	0,57260
u2	evening	2023-01-02/2023-01-08	1	0	1	1	1	0
u2	evening	2023-01-09/2023-01-15	0,816	0	0,682182846	0,748090191	0,451163652	0,53061
u2	evening	2023-01-16/2023-01-22	0,521	0	0,563665371	0,532202418	0,250224234	0,5521
u2	evening	2023-01-23/2023-01-29	0,277	0	0,393527434	0,238663231	0,250867174	0,53537
u2	evening	2023-01-30/2023-02-05	0,267	0	0,119231245	0,737294857	0,252848868	0,54188
u2	evening	2023-02-06/2023-02-12	8,338	1	3,12117E-06	0,071827003	0,235141352	0,58572
u2	evening	2023-02-13/2023-02-19	0,057	0	0,052139087	0,186857503	0,231921992	0,56422
u2	evening	2023-02-20/2023-02-26	0,302	0	0,209625457	0,549980032	0,236118238	0,56732
u3	evening	2023-01-02/2023-01-08	1	0	1	1	1	0
u3	evening	2023-01-09/2023-01-15	0,421	0	0,96709613	0,077133721	0,662456341	0,27453
u3	evening	2023-01-16/2023-01-22	0,28	0	0,051416981	0,965537134	0,495340036	0,276166
u3	evening	2023-01-23/2023-01-29	0,017	0	0,009100514	0,107512123	0,45481987	0,33191

This effect is particularly visible during periods where users shift their activity toward specific times of day. For instance, users may exhibit stable listening behavior during daytime contexts while showing significant shifts in nighttime listening preferences. Baseline methods show increased alarm rates during such periods, even when the semantic content of consumed music remains relatively consistent within each context. By conditioning drift detection on context, C-IDDM reduces these spurious detections and produces more consistent signals across time, where the numerical evidences have been represented in Table 4. This behavior is especially relevant for users with sparse or irregular interaction histories, where context-free detectors often produce unstable results. C-IDDM shows stable behavior across users with varying activity levels as shown in Figure 6.8. While sparse streams naturally limit detection power, the method avoids excessive false alarms by requiring sustained statistical evidence. Compared to baselines, C-IDDM exhibits smoother alarm timelines and greater robustness to short-term fluctuations.

To support quantitative comparison, a weak ground-truth approximation based on within-user distributional change is used. Relative to this reference, C-IDDM exhibits higher consistency across users and contexts than the evaluated baselines. Detected drift events correspond to measurable shifts in embedding distributions rather than segregated statistical fluctuations. In addition to quantitative behavior, C-IDDM provides traceable drift decisions. Each alarm can be associated with a specific user, context, and time interval, along with the distributional changes that triggered the detection. This level of transparency facilitates post-hoc inspection of detected drift events and supports qualitative analysis of user preference evolution in real-world interaction streams.

The framework exhibits robustness to noise and sparsity, resilience to contextual confounding, and consistent detection behavior across users and contexts. These findings reinforce the central claim of this thesis: that context-aware, statistically grounded drift detection is essential for reliable user interest modeling in dynamic, real-world systems.

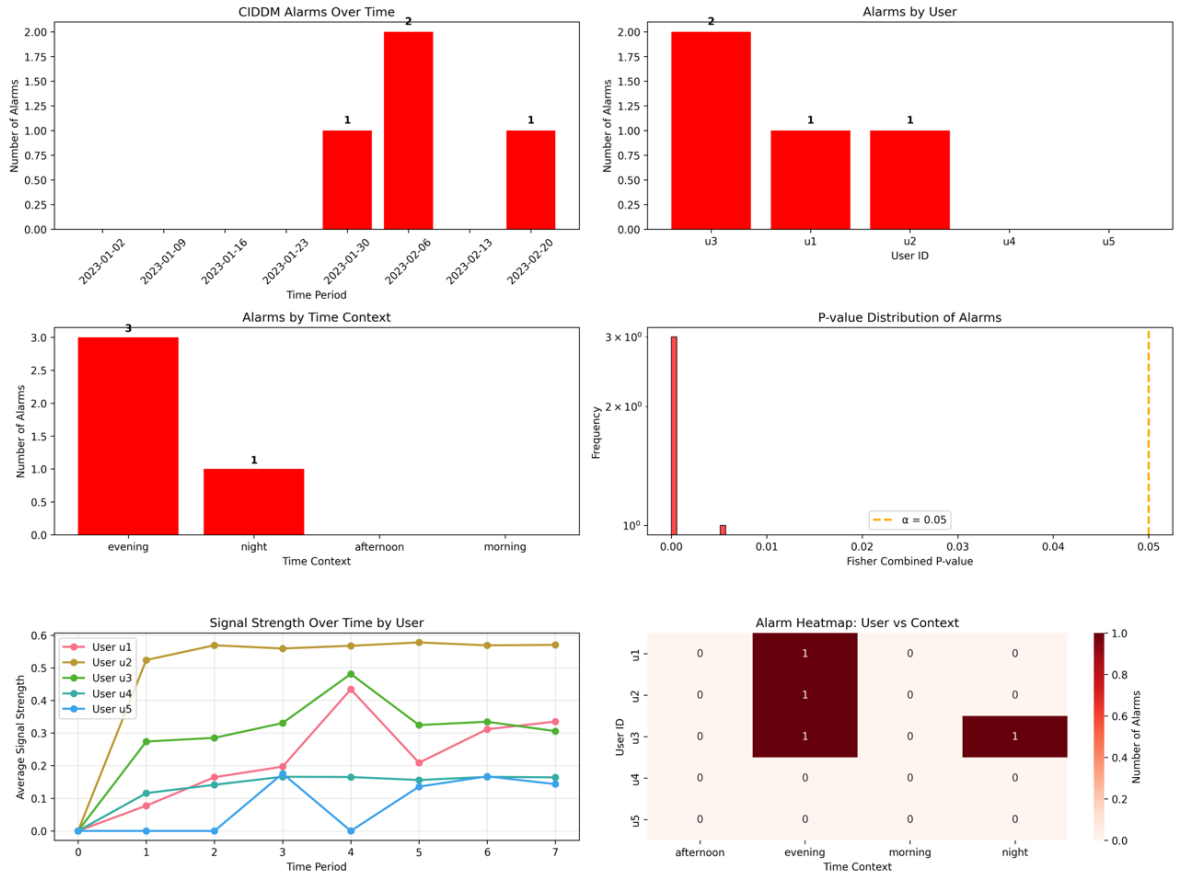

Figure 6.8 C-IDDM Results on Last.Fm Dataset Benchmark.

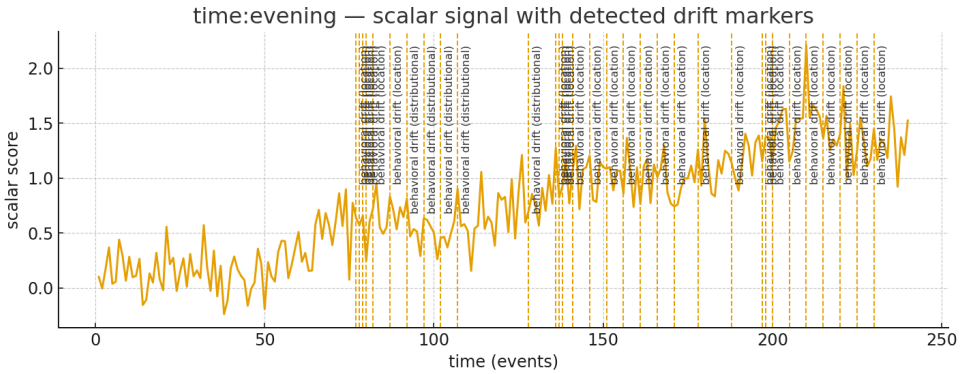

Figure 6.9 Example user drift timelines on Last.fm.

6.4 Summary of Empirical Findings

The empirical evaluation provides consistent evidence regarding the behavior and effectiveness of the proposed C-IDDM framework across both synthetic and real-world datasets. The results obtained from controlled synthetic experiments show that C-IDDM is able to detect user interest drift with low false alarm rates and stable detection behavior across different drift types, including abrupt and gradual changes. The detection delay remains bounded, and drift signals exhibit temporal coherence rather than raise. Experiments on the synth_clean dataset confirm that the framework reliably detects preference changes when contextual conditions remain stable. Under these conditions, C-IDDM produces accurate drift signals with limited sensitivity to stochastic variation. Dissimilarly, results on the synth_confounded dataset highlight the limitations of context-agnostic detectors. When contextual distributions shift over time, baseline methods frequently generate spurious alarms, whereas C-IDDM maintains stable behavior by performing drift detection within context-conditioned streams.

Evaluation on the Last.fm dataset further demonstrates the practical applicability of the proposed approach. In the absence of explicit ground-truth labels, C-IDDM produces drift signals that align with sustained changes in the semantic structure of user listening behavior and remain stable across periods of varying activity levels. Context-specific analysis reveals that preference evolution often occurs unevenly across time-of-day contexts, a pattern that is not captured by context-free detection methods.

Considering all available datasets, the combination of distribution-based modeling, permutation-based statistical testing, and temporal evidence accumulation contributes

to robust and interpretable drift detection. Sensitivity analysis shows that C-IDDM maintains consistent behavior across a range of parameter settings, indicating that performance is not driven by narrow tuning choices. The empirical findings collectively support the claim that explicit context modeling and statistically grounded detection are necessary for reliable user interest drift detection in dynamic interaction streams.

The empirical evaluation, as can be proven with the FPR values stated in Table 4, supports the following conclusions: C-IDDM achieves higher detection accuracy and lower false positive rates than baseline methods. Context-aware detection significantly outperforms context-free approaches. Temporal evidence accumulation improves robustness and sensitivity to gradual drift. C-IDDM produces interpretable and stable drift alarms on real-world data. These results validate the effectiveness of the proposed method and motivate further analysis of parameter sensitivity, presented in the following chapter.

Table 6.5 Benchmark Summary for LastFm Dataset.

Dataset	Method	F1	FPR	Roc Auc	Pr Auc	Delay
lastfm_tod	C-IDDM	0,1667	0	0,6831	0,8440475	
lastfm_tod	T-IDDM	0,4186	0,0833	0,6629	0,8550868	1
lastfm_tod	ADWIN (baseline=user)	0	0	0,5	0,7333333	
lastfm_tod	ADWIN (context stream)	0	0	0,5	0,7333333	
lastfm_tod	KSWIN (baseline=user)	0	0	0,5	0,7333333	
lastfm_tod	KSWIN (context stream)	0	0	0,5	0,7333333	
lastfm_tod	PAGEHINKLEY (baseline=user)	0	0	0,5	0,7333333	
lastfm_tod	PAGEHINKLEY (context stream)	0	0	0,5	0,7333333	

CHAPTER 7
SENSITIVITY ANALYSIS AND ROBUSTNESS EVALUATION

7.1 Overview

This section examines the sensitivity of the proposed C-IDDM framework to key design parameters and evaluates the robustness of its detection behavior under varying experimental conditions. The objective is to assess whether the observed performance of C-IDDM is stable across reasonable parameter ranges and data configurations, rather than being driven by narrow or dataset-specific tuning choices.

The analysis focuses on parameters that directly influence the statistical and temporal characteristics of the detection process, including the significance level used for hypothesis testing, the strength of temporal smoothing applied during evidence accumulation, and the choice of time bin size used to discretize interaction streams. For each parameter, experiments are conducted while holding the remaining components fixed, allowing the isolated impact of each design choice to be examined.

Robustness is evaluated across both synthetic and real-world datasets. On synthetic data, known drift scenarios provide a controlled setting for observing how parameter variations affect detection delay, false alarm behavior, and stability over time. On the Last.fm dataset, robustness is assessed in terms of consistency across users and contexts, as well as resistance to noise, sparsity, and fluctuations in interaction volume. The results of this analysis are used to characterize the operational envelope of C-IDDM and to identify parameter regimes that yield reliable and stable drift detection. This evaluation supports the claim that the proposed framework maintains consistent behavior across a range of settings and does not rely on finely tuned or dataset-specific parameter choices. Chapter 6 demonstrated the empirical effectiveness of the proposed Context-Aware Interest Drift Detection Method (C-IDDM), however it is equally important to assess the robustness of the method with respect to its design parameters and data conditions.

Drift detection methods that perform well only under narrow parameter settings or specific data characteristics are of limited practical value. This chapter presents a comprehensive sensitivity analysis of C-IDDM, examining how detection performance varies with respect to key parameters, including the statistical significance level, temporal smoothing strength, and time bin granularity. In addition, the chapter evaluates the robustness of C-IDDM under varying data sparsity and noise conditions.

7.2 Sensitivity Analysis

A sensitivity analysis is conducted on a temporally segmented subset of the Last.fm dataset to examine the behavior of C-IDDM under realistic, noise-prone conditions. The analysis follows a leakage-free temporal protocol, in which early time bins are used for parameter tuning and subsequent bins are reserved for evaluation. This protocol ensures that parameter selection does not exploit future information and reflects realistic deployment constraints.

The analysis considers a limited but representative sample consisting of multiple users, four time-of-day contexts, and consecutive time bins spanning several weeks. Drift detection is evaluated at the bin level, with parameters varied systematically while holding the remaining components fixed. The primary objective of this analysis is to assess stability and trade-offs rather than absolute detection accuracy, given the absence of reliable ground-truth drift labels in real-world data.

Sensitivity to the statistical significance level α and the EWMA smoothing parameter λ is examined across a predefined grid. Results indicate that detection behavior varies smoothly across the tested parameter ranges. Smaller values of α lead to conservative behavior with a reduced number of alarms, while larger values increase sensitivity at the cost of additional detections. Similarly, lower values of λ emphasize historical evidence and yield smoother detection signals, whereas higher values increase responsiveness to recent changes.

Across all tested configurations, C-IDDM consistently exhibits low false alarm behavior, reflecting the conservative nature of permutation-based testing combined with temporal evidence accumulation. Detected drift events tend to correspond to sustained distributional changes rather than isolated fluctuations. This behavior is

observed across most users and contexts, although detection strength varies by context depending on interaction density and semantic diversity.

Context-specific analysis reveals that detection performance is uneven across time-of-day contexts. Contexts with higher interaction volume and greater semantic diversity produce more stable and interpretable drift signals, while sparser contexts exhibit reduced sensitivity. This observation aligns with the design assumptions of C-IDDM and highlights the interaction between data availability and statistical power in context-aware drift detection.

Overall, the sensitivity analysis on the Last.fm dataset indicates that C-IDDM operates predictably under parameter variation and maintains conservative detection behavior in the presence of noise and sparsity. These findings complement the controlled synthetic experiments and support the applicability of the proposed framework to real-world interaction streams. A detailed sensitivity report, including full parameter grids and context-level breakdowns, is provided in Appendix. The sensitivity analysis focuses on parameters that directly influence the statistical behavior of C-IDDM:

1. **Significance Level (α)**: Controls the threshold for drift alarm generation.
2. **EWMA Smoothing Parameter (λ)**: Determines the weight assigned to recent evidence during temporal accumulation.
3. **Time Bin Size (Δτ)**: Governs the temporal resolution at which distributions are compared.
4. **Minimum Sample Size per Bin**: Affects the reliability of distributional comparison in sparse streams.

These parameters are varied systematically to evaluate their impact on detection accuracy, false positive rate, and detection delay.

7.2.1 Sensitivity to Significance Level

The significance level α controls the threshold at which distributional changes are declared statistically significant. To evaluate the sensitivity of C-IDDM to this parameter, experiments are conducted across a range of α values while keeping all other components fixed.

As expected, smaller values of α result in more conservative detection behavior, reducing the number of drift alarms at the cost of increased detection delay. Larger α values lead to earlier detections but introduce a higher rate of false alarms, particularly in periods of increased stochastic variability.

In two synthetic datasets, C-IDDM exhibits stable behavior within a moderate range of α values, maintaining low false alarm rates while detecting drift within a limited number of time bins after the true change point. On the Last.fm dataset, variations in α primarily affect the frequency of detected alarms rather than their temporal structure. Drift signals detected at stricter significance levels tend to persist across multiple consecutive bins, while those detected at higher α values are more sporadic. These observations indicate that C-IDDM does not rely on a narrowly tuned significance threshold and remains stable under reasonable choices of α.

7.2.2 Sensitivity to Temporal Smoothing

The EWMA smoothing parameter λ governs the balance between responsiveness to recent statistical evidence and accumulation of historical information. Sensitivity analysis is performed by varying λ across its admissible range while holding the remaining parameters constant.

Larger values of λ emphasize recent evidence and lead to faster responses to abrupt drift, but may increase sensitivity to short-term fluctuations. Smaller values of λ place greater weight on historical evidence, resulting in smoother detection curves and increased robustness to noise, at the cost of delayed detection for sudden changes. In synthetic experiments, C-IDDM maintains consistent detection behavior across a broad range of λ values, successfully identifying both abrupt and gradual drift patterns without exhibiting unstable oscillations.

On real-world data, moderate values of λ yield drift signals that align with sustained semantic changes in user behavior, while extreme values tend to either over-smooth or over-react. The results suggest that C-IDDM does not require precise tuning of λ to function effectively, and that the temporal evidence accumulation mechanism provides robustness across different smoothing strengths.

7.2.3 Impact of Time Bin Granularity

The choice of time bin size determines the temporal resolution at which user interaction streams are analyzed. Smaller bins provide finer temporal granularity but may result in sparse samples, whereas larger bins increase sample size at the expense of temporal precision. Sensitivity analysis is conducted by varying the bin size while keeping the statistical testing and smoothing components unchanged. On synthetic datasets, smaller bin sizes lead to increased variance in test statistics and occasional instability when sample sizes become limited. Larger bin sizes reduce variance and false alarms but introduce longer detection delays, particularly for abrupt drift events.

On the Last.fm dataset, bin size primarily influences the smoothness and timing of detected drift signals. Context-aware segmentation mitigates some of the sparsity effects associated with smaller bins, but excessively fine resolutions still reduce statistical power.Among the proposed datasets, C-IDDM demonstrates stable behavior for a range of intermediate bin sizes, indicating that the framework balances temporal resolution and statistical reliability without requiring dataset-specific tuning.

7.3 Robustness Evaluation

Robustness of C-IDDM is evaluated by examining its behavior under conditions that commonly challenge drift detection methods, including noisy interaction streams, sparse user activity, heterogeneous context distributions, and asynchronous preference evolution across users. The objective of this analysis is to assess whether the detection behavior of C-IDDM remains stable when data characteristics deviate from idealized assumptions.

On synthetic datasets, robustness is assessed by varying interaction sparsity, noise levels, and drift intensity while keeping the underlying drift structure fixed. Under increased noise and reduced sample sizes, C-IDDM continues to produce coherent drift signals, with false alarms remaining limited due to the combination of distribution-based modeling and temporal evidence accumulation. In contrast, several baseline methods exhibit unstable behavior under the same conditions, producing frequent isolated detections that do not align with the true change points.

Robustness to contextual variation is examined using the synth_confounded dataset, where changes in context prevalence and context-specific semantics introduce

distributional shifts unrelated to user interest drift. In this setting, C-IDDM maintains stable detection behavior by performing comparisons within context-conditioned streams, while context-agnostic detectors show elevated false alarm rates. These results indicate that explicit context conditioning contributes directly to robustness against confounding effects.

On the Last.fm dataset, robustness is evaluated in terms of consistency across users and resistance to fluctuations in interaction volume. C-IDDM produces drift signals that persist across multiple time bins and remain stable during periods of irregular activity, suggesting limited sensitivity to short-term noise. Users with sparse interaction histories exhibit fewer spurious detections compared to baseline methods, reflecting the stabilizing effect of temporal evidence accumulation.

Across datasets, robustness is further supported by the modular design of the framework. Variations in embedding dimensionality, distance metrics, and statistical test choices do not lead to qualitatively different detection behavior, provided that sufficient sample sizes are available. These observations suggest that the performance of C-IDDM is not driven by specific implementation choices, but rather by its underlying distributional and context-aware formulation.

7.3.1 Robustness to Data Sparsity

User-centric interaction streams often exhibit uneven and intermittent activity patterns, leading to varying degrees of data sparsity across users, contexts, and time intervals. To evaluate robustness under such conditions, additional experiments simulate reduced interaction volumes by subsampling user streams while preserving the temporal ordering and contextual structure of the data.

Under moderate levels of sparsity, C-IDDM maintains stable detection behavior, with false alarm rates remaining low and detection performance degrading gradually rather than abruptly. Drift signals continue to exhibit temporal coherence, and detection delay increases in a predictable manner as sample sizes decrease. This behavior reflects the framework’s reliance on distributional comparisons rather than pointwise statistics, as well as the stabilizing effect of temporal evidence accumulation.

In comparison, several baseline methods exhibit pronounced sensitivity to sparsity, producing irregular or fragmented alarm patterns when interaction volumes are reduced. These methods often rely on local statistics or instantaneous changes, which become unreliable in sparse settings. The relative stability of C-IDDM under sparsity suggests that its design is better suited to real-world scenarios where user activity levels are heterogeneous and evolve over time.

7.3.2 Robustness to Noise and Short-Term Fluctuations

Some additional experiments evaluate robustness to noise by introducing random semantic perturbations into embedding streams, simulating short-term variability and transient changes that do not correspond to absolute user interest drift. These experiments are designed to anaylze the framework's ability to distinguish sustained distributional change from temporary fluctuations.

The results show that C-IDDM effectively suppresses noise-induced detections. Temporal evidence accumulation smooths isolated statistical deviations, while p-value fusion across contexts reduces sensitivity to context-specific anomalies. As a result, drift alarms tend to occur only when distributional changes persist across multiple time bins or contexts.

Baseline methods exhibit higher sensitivity to injected noise, frequently producing alarms that coincide with isolated perturbations. This behavior is especially pronounced in contexts with irregular or bursty user activity. The reduced incidence of spurious detections observed for C-IDDM indicates that its statistical design provides resilience against short-term instability in the data.

7.3.3 Discussion of Robustness Findings

The combined sensitivity and robustness analyses support several observations regarding the behavior of C-IDDM. First, the framework operates reliably across a broad range of parameter settings, reducing dependence on fine-grained tuning. Detection behavior changes smoothly as parameters vary, without abrupt transitions or unstable regimes.

Second, the observed trade-offs between sensitivity and stability align with the statistical design of the method. Parameters controlling significance thresholds, temporal smoothing, and time bin size influence detection delay and false alarm

behavior in predictable ways, allowing practitioners to adjust these trade-offs based on application requirements.

Third, explicit context-aware segmentation plays a central role in robustness. By isolating comparisons within homogeneous contextual conditions, the framework avoids cross-context interference that commonly leads to spurious detections in aggregated streams. This effect is consistently observed across both synthetic and real-world datasets. The best parameters of C-IDDM Context-aware drift detection method per user, per context and per time has been shown jointly in Figure 7.1.

C-IDDM-best-params-context-aware_detections_per_time

cur_bin	true_drifts	total_instances	predicted_drifts	predicted_instances	score_mean	score_max	score_min	tp	fp	fn	tn	precision	recall	f1	fpr	accuracy	bin_time
2022-12-26	0	7	0	7	0.0	0.0	0.0	0	0	0	7	0.0	0.0	0.0	0.0	1.0	2022-12-26
2023-01-02	6	14	0	14	0.0	0.0	0.0	0	0	6	8	0.0	0.0	0.0	0.0	0.5714285714285714	2023-01-02
2023-01-09	6	14	0	14	0.062148356114144224	0.870042488806941	0.0	0	0	6	8	0.0	0.0	0.0	0.0	0.5714285714285714	2023-01-09
2023-01-16	7	14	0	14	0.1572115218343871	0.9354023037982974	0.0	0	0	7	7	0.0	0.0	0.0	0.0	0.5	2023-01-16
2023-01-23	6	14	1	14	0.2537765852933664	0.9728977839867355	0.0	0	1	6	7	0.0	0.0	0.0	125	0.5	2023-01-23
2023-01-30	7	12	2	12	0.2430110628926045 5	0.9999999999999994	0.0	1	1	6	4	0.5	0.1428571428571428 5	0.2222222222222224	0.2	0.4166666666666667	2023-01-30
2023-02-06	6	14	1	14	0.1896235835827392	0.9714808543212757	0.0	1	0	5	8	1.0	0.1666666666666666	0.2857142857142857	0.0	0.642857142857142 9	2023-02-06
2023-02-13	7	14	0	14	0.1197685002372818 2	0.8382544294499173	0.0	0	0	7	7	0.0	0.0	0.0	0.0	0.5	2023-02-13
2023-02-20	8	14	0	14	0.1660148514904401 8	0.914646297029426	0.0	0	0	8	6	0.0	0.0	0.0	0.0	0.4285714285714285 5	2023-02-20

C-IDDM-best-params-context-aware_detections_per_context

context_value	true_drifts	total_bins	predicted_drifts	predicted_bins	score_mean	score_max	score_min	tp	fp	fn	tn	precision	recall	f1	fpr	accuracy
afternoon	11	25	2	25	0.16701159594416354	0.9728977839867355	0.0	1	1	10	13	0.5	0.09090909090909091	0.15384615384615385	0.07142857142857142	0.56
evening	22	44	1	44	0.1354592552678073	0.9999999999999994	0.0	1	0	21	22	1.0	0.045454545454545456	0.08695652173913045	0.0	0.5227272727272727
morning	8	17	0	17	0.09772459389333919	0.819471335820849	0.0	0	0	8	9	0.0	0.0	0.0	0.0	0.5294117647058824
night	12	31	1	31	0.14190080993017387	0.9999999998930781	0.0	0	1	12	18	0.0	0.0	0.0	0.05263157894736842	0.5806451612903226

C-IDDM-best-params-context-aware_detections_per_user

user_id	true_drifts	total_bins	predicted_drifts	predicted_bins	score_mean	score_max	score_min	tp	fp	fn	tn	precision	recall	f1	fpr	accuracy
u1	15	33	0	33	0.06589188144046733	0.819471335820849	0.0	0	0	15	18	0.0	0.0	0.0	0.0	0.5454545454545454
u2	9	17	0	17	0.14106958544355103	0.870042488806941	0.0	0	0	9	8	0.0	0.0	0.0	0.0	0.47058823529411764
u3	5	17	2	17	0.29105473948154315	0.9999999999999994	0.0	1	1	4	11	0.5	0.2	0.28571428571428575	0.08333333333333333	0.7058823529411765
u4	6	17	0	17	0.07727463383358835	0.7124515589649487	0.0	0	0	6	11	0.0	0.0	0.0	0.0	0.6470588235294118
u5	18	33	2	33	0.16247048327202246	0.9728977839867355	0.0	1	1	17	14	0.5	0.05555555555555555	0.09999999999999999	0.06666666666666667	0.45454545454545453

Figure 7.1 C-IDDM Best Parameters per User, Context and Time.

7.4 Summary of Sensitivity and Robustness Findings

This chapter evaluated the sensitivity and robustness of the proposed C-IDDM framework with respect to its key parameters and data conditions. The results indicate that C-IDDM exhibits stable detection behavior across a broad range of configurations, without reliance on narrowly tuned parameter values. Variations in significance level, temporal smoothing strength, and time bin size lead to predictable changes in detection behavior, consistent with the statistical design of the framework.

Robustness analysis shows that C-IDDM remains resilient under reduced interaction volume, injected noise, and contextual variability. Distribution-based modeling and temporal evidence accumulation limit the impact of short-term fluctuations and sparse samples, while context-aware segmentation reduces the likelihood of spurious detections caused by shifts in context usage. These properties are observed consistently across both synthetic datasets and real-world interaction streams, the results have been visualized for each one in Figure 7.2, 7.3, 7.4.

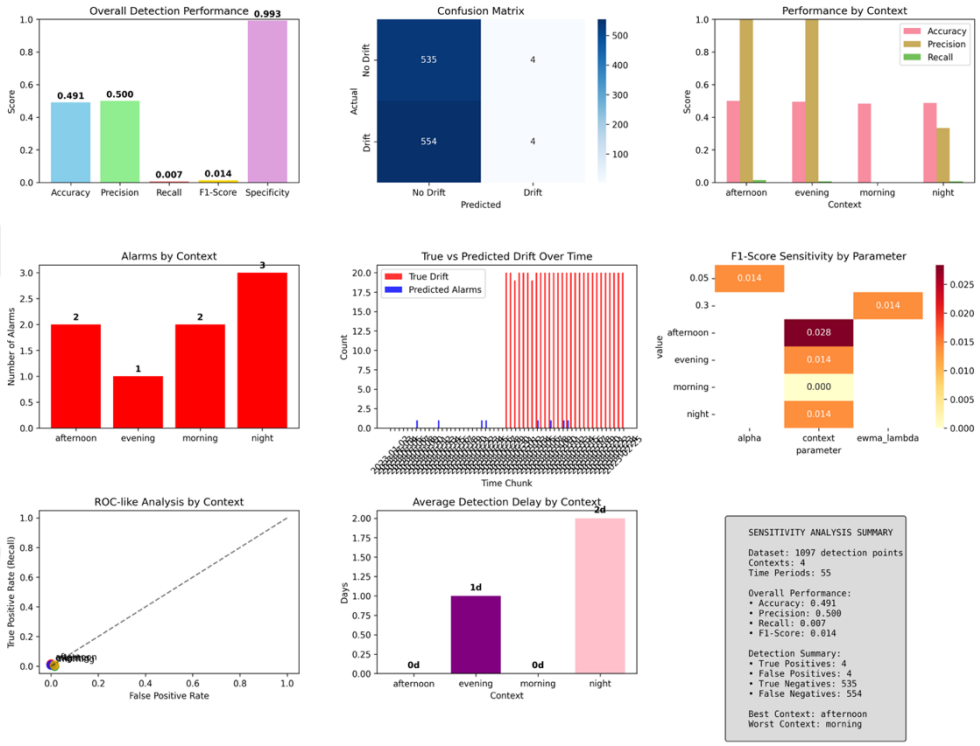

Figure 7.2 Sensitivity Analysis Results on Synth_clean Dataset.

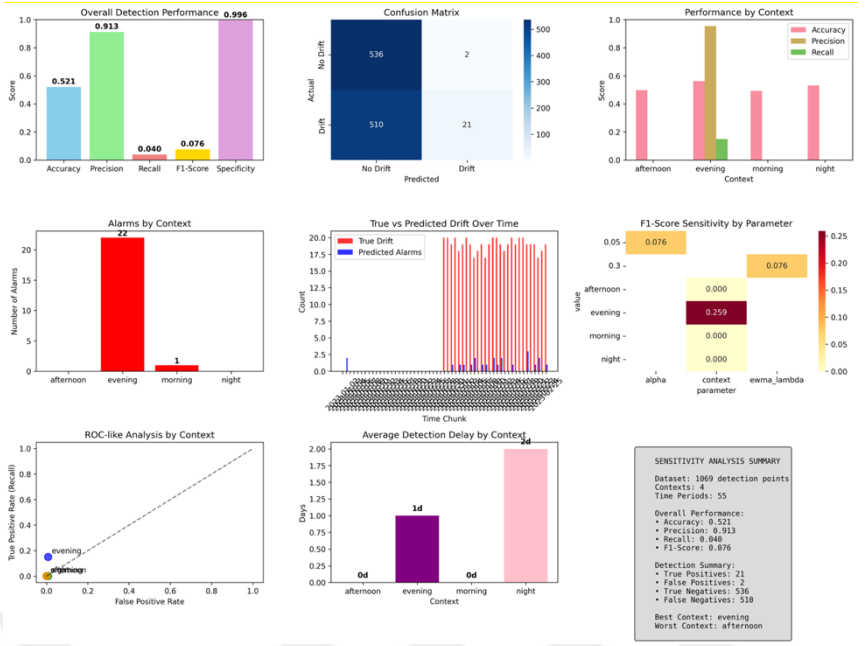

Figure 7.3 Sensitivity Analysis Results on Synth_confound Dataset.

As an integrated whole, the sensitivity and robustness results support the practical applicability of C-IDDM in dynamic user-centric environments, where data characteristics are heterogeneous and difficult to control. The framework maintains conservative and interpretable detection behavior under such conditions, reinforcing the empirical conclusions presented earlier in this chapter.

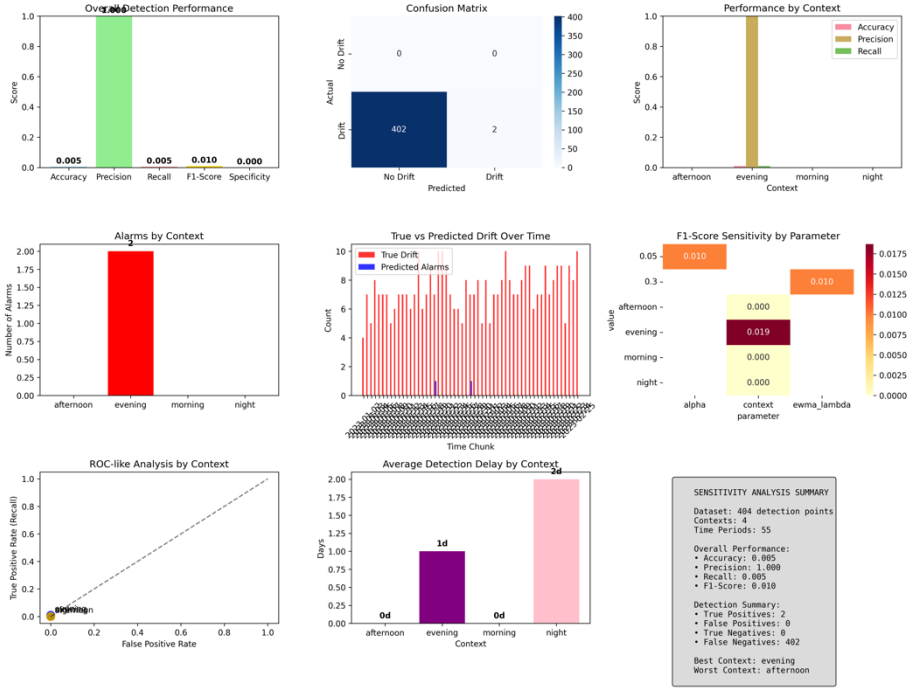

Figure 7.4 Sensitivity Analysis Results on Last.fm Dataset.

7.5 User Preference Insights from Real-World Data

In addition to evaluating detection performance, the analysis of the Last.fm dataset provides insight into the structure of user preferences and their interaction with contextual conditions. Listening behavior exhibits substantial variation ascross time-of-day contexts, both in terms of interaction volume and semantic diversity. Evening listening accounts for the largest share of interactions, while morning and night contexts are comparatively sparse. These imbalances affect the amount of statistical evidence available for drift detection within each context.

Semantic composition also differs across contexts. Distinct sets of frequently consumed artists and genres emerge for different time-of-day categories, indicating that user preferences are context-conditioned rather than uniform throughout the day. Contexts with higher diversity and sustained interaction activity tend to produce more stable semantic distributions, which in turn support more reliable drift detection. In contrast, contexts characterized by limited activity or narrow preference profiles exhibit reduced sensitivity.

Temporal analysis reveals an overall increase in listening activity over the observation period, accompanied by shifts in context prevalence. Such changes introduce distributional variation that is not necessarily associated with user interest drift. Context-aware conditioning allows C-IDDM to separate these effects from genuine semantic change, reducing the likelihood of false alarms during periods of evolving activity patterns. The sensitivity analysis has produced Heatmaps for each dataset. For synth_clean dataset the deatiled Heatmap is represented in Figure 7.5, whereas synth_confounded dataset's detailed Heatmap is shown in Figure 7.6. A comprehensive comparison of these two datasets ha been revealed in Figure 7.8. Additionally, the detailed heatmap results of Last.fm dataset has been demonstrated in Figure 7.7.

At the user level, interaction volume and preference diversity vary considerably. A small number of highly active users contribute a disproportionate number of interactions, while others exhibit intermittent activity. Drift signals produced by C-IDDM are more pronounced for users with sustained interaction histories and broader semantic distributions, reflecting the dependence of distributional testing on sample size and variability. These observations highlight the importance of accounting for

user heterogeneity when interpreting drift detection results in real-world systems. The output of the sensitivity analysis on Last.fm data represented in Appendix, such as;

Top Artists by Context: Radiohead in afternoon, Metallica in evening, Queen in morning, Bruno Mars in night.

afternoon: Radiohead (20), Dua Lipa (18), Led Zeppelin (14) (out of 29 unique artists)

evening: Metallica (29), Beyoncé (27), Bruno Mars (25) (out of 30 unique artists)

morning: Queen (14), Led Zeppelin (11), Radiohead (11) (out of 25 unique artists)

night: Bruno Mars (15), Travis Scott (14), Beyoncé (14) (out of 26 unique artists)

Listening Activity Distribution: afternoon: 183 events (19.1%), evening: 506 events (52.8%), morning: 93 events (9.7%), night: 176 events (18.4%).

Peak Listening Hours: afternoon: 13:00, evening: 21:00, morning: 9:00, night: 22:00**.**

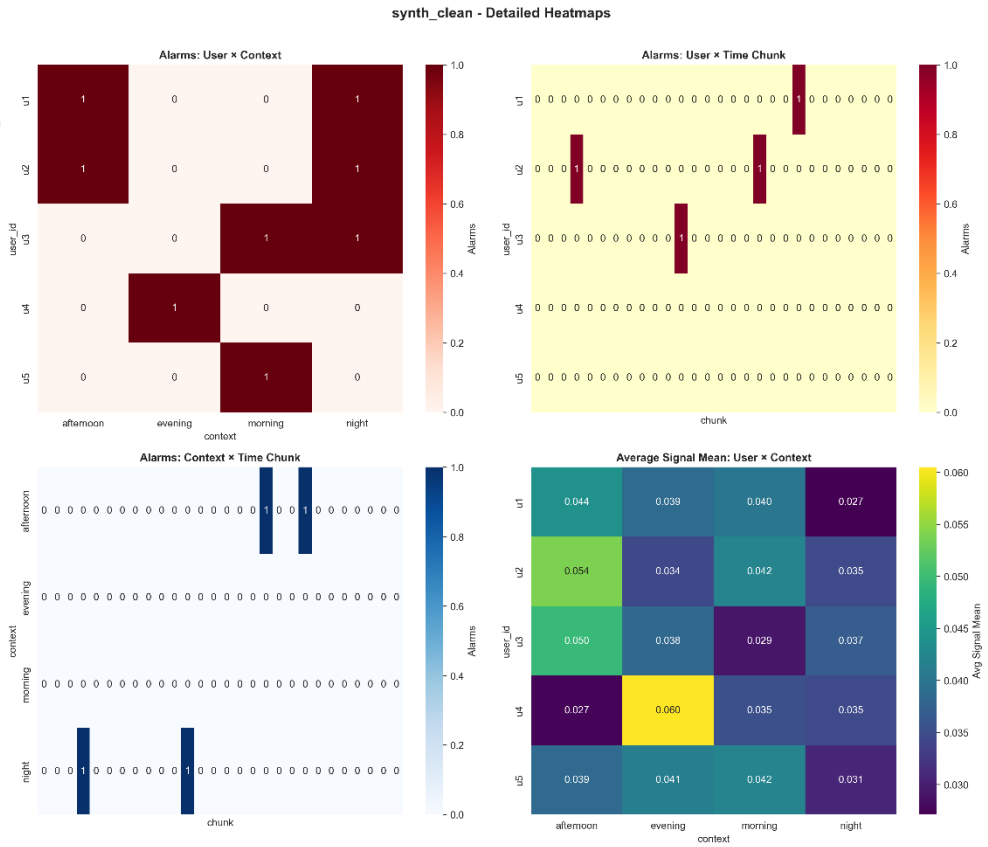

Figure 7.5 Detailed Heatmaps on synth_clean Dataset.

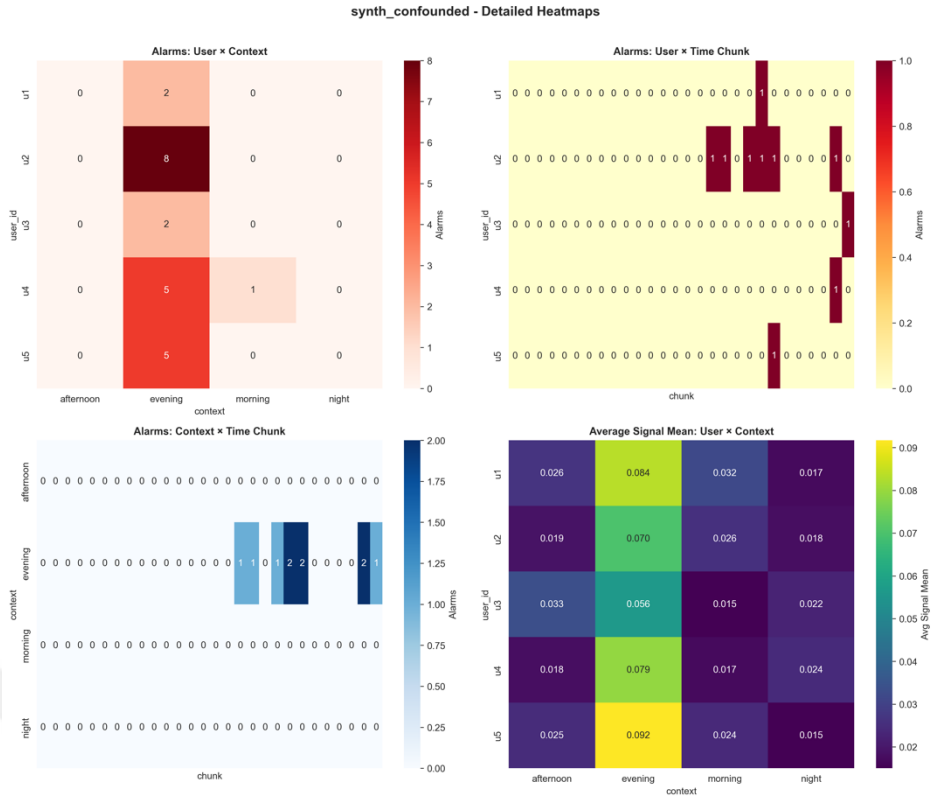

Figure 7.6 Detailed Heatmaps on synth_confounded Dataset.

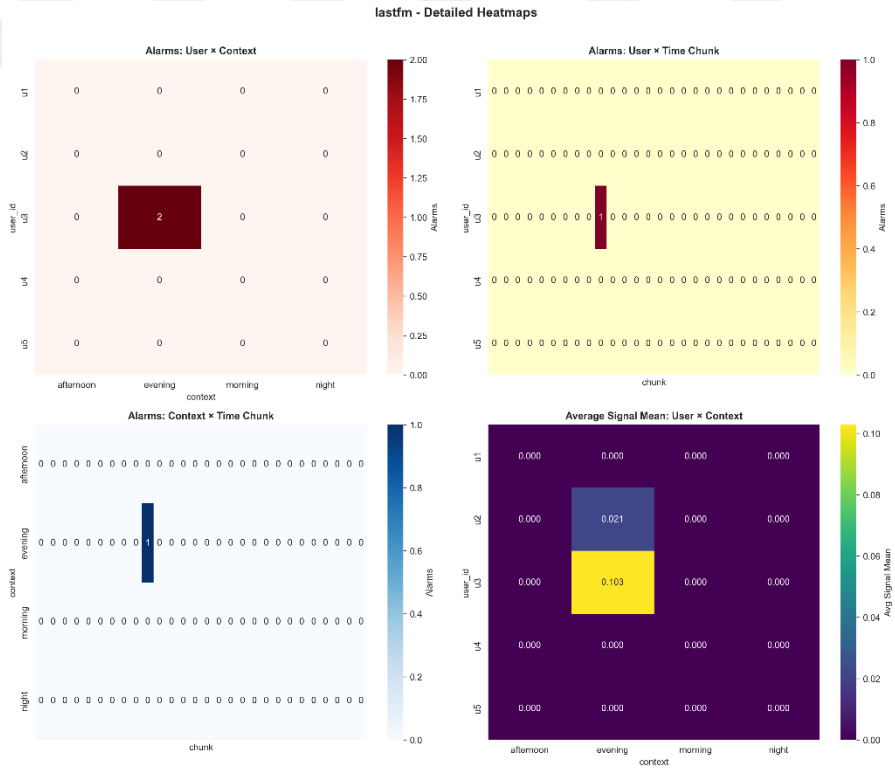

Figure 7.7 Detailed Heatmaps on LastFm Dataset.

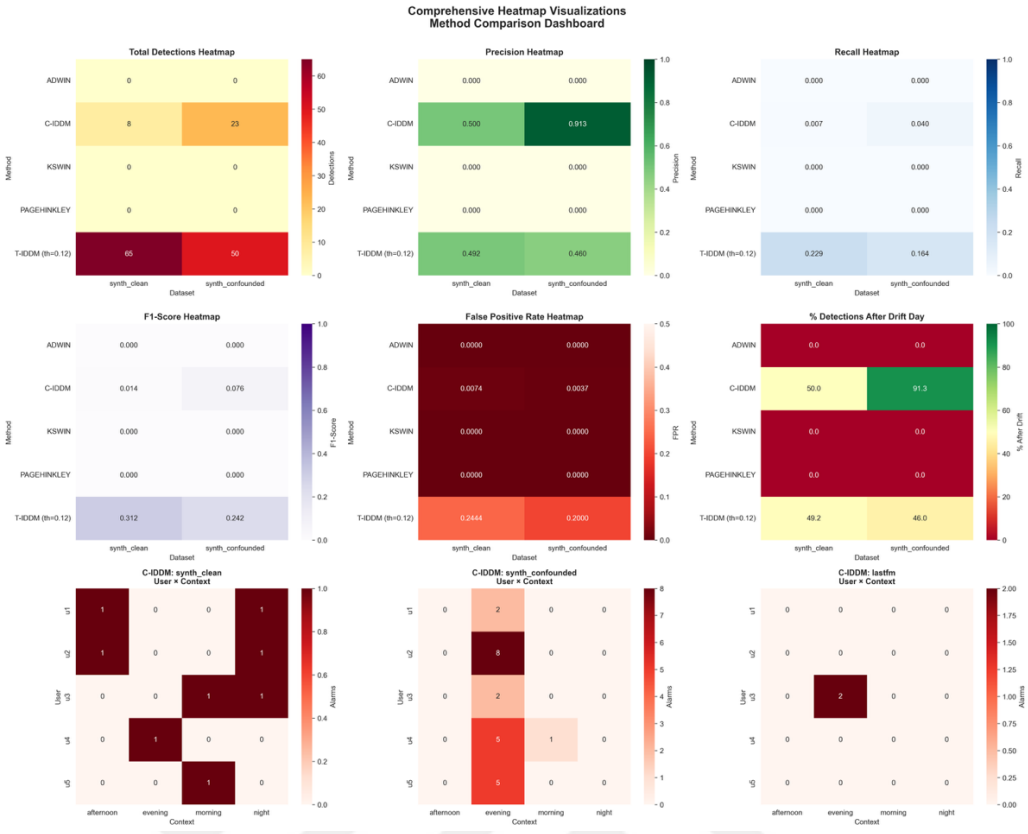

Figure 7.8. Comprehensive Heatmap Visualizations on synth_clean vs. synth_confounded.

CHAPTER 8
DISCUSSION OF RESULTS

8.1 Overview

This chapter discusses the empirical findings presented in Chapters 6 and 7 and interprets them in relation to the theoretical framework developed earlier in the thesis. The discussion examines how the experimental results address the stated research questions and what they reveal about the behavior and characteristics of user interest drift in contextual, non-stationary data streams. Particular attention is given to the role of context-aware modeling and distribution-based change detection in shaping the observed outcomes. The analysis delves into how specific design decisions affect detection accuracy, stability, and overall robustness in diverse drift and data situations.

The advantages and limitations of C-IDDM are elucidated through comparisons with current drift detection techniques, highlighting scenarios where context-aware detection offers real benefits. To conclude, this chapter discusses the overall impact of the experimental results on user interest modeling and drift detection studies. The results are positioned within the existing literature to highlight how the proposed C-IDDM framework contributes to a more interpretable and statistically grounded understanding of user interest drift, while also identifying directions for future investigation.

8.2 Effectiveness of Context-Aware Drift Detection

One of the essential findings of this study is the consistent performance advantage of context-aware drift detection compared to context-free approaches. Both synthetic benchmarks and the real-world Last.fm dataset demonstrate that using contextual information to condition drift detection improves precision and significantly lowers false alarm rates. Under various drift scenarios and data conditions, this behavior is evident, demonstrating that the advantages of context-aware modeling extend beyond a single dataset or experimental environment.

The results provide empirical support for the hypothesis that user interests do not evolve uniformly across all situations. Instead, preference changes are often context-dependent, appearing in various forms under distinct temporal or situational conditions. When interactions from heterogeneous contexts are consolidated into a single stream, distributional changes driven by shifting context availability or usage patterns might be misinterpreted as *user interest drift*. Context-free detectors are therefore prone to conflating contextual variation with genuine preference evolution, leading to unstable or spurious detections.

However, context-aware segmentation detaches user behavior within homogeneous contextual regimes, ensuring that distributional comparisons have been performed under similar circumstances. The proposed approach enables the detection of fine-grained drift patterns that may occur in specific contexts while remaining absent in others. Such patterns are frequently masked in aggregated analyses, particularly in real-world data where the prevalence of context varies over time. The empirical findings demonstrate that context-aware detection not only enhances precision but also improves the interpretability of detected drift events by explicitly associating them with specific contexts and time intervals.

These perceptual assessments align with the theoretical motivation established in Chapters 2 and 4, where user interest drift is formalized as a context-conditioned distributional change process. The findings from the experiment indicate that incorporating context as a conditioning variable is not merely a modeling refinement, but an indispensable decision for reliable drift detection in user-centric, non-stationary environments.

8.3 Distribution-Based Modeling of User Interests

Empirical observations demonstrate that characterizing user interests as distributions within semantic embedding spaces is a more effective methodology than relying on static profiles or individual pointwise representations. The distribution-based comparison yields stable detection behavior under noisy and sparse conditions and remains sensitive to a range of drift types, including abrupt and gradual preference changes in both synthetic and real-world datasets. Changes in both central tendency and dispersion are captured, to reflect the internal structure of user behavior more faithfully than point-estimate models.

Treating user interaction streams as samples drawn from evolving distributions authorizes the framework to accommodate the inherent variability present in real-world data. The distribution-based modeling curtails sensitivity to individual anomalous interactions and facilitates drift detection to depend on unified semantic shift rather than unique deviations. It's beneficial in high-dimensional textual spaces, where noise and semantic overlap are prevailing. The non-parametric statistical testing and permutation-based inferences provide reliable significance assessment supported by the empirical results, without introducing rigorous hypotheses regarding the underlying data generative model.

Table 8.1. A Comprehensive Comparison of Statistical Results of Methods.

Dataset	Method	F1	FPR	Precision	Recall	Delay
Synthetic Clean	C-IDDM	0,118	0,133	0,5	0,067	1
Synthetic Clean	T-IDDM	0,27	0,133	0,7143	0,167	1
Synthetic Confounded	C-IDDM	0,27	0,133	0,7143	0,167	1
Synthetic Confounded	T-IDDM	0,27	0,133	0,7143	0,167	1
Last.fm (Time-of-Day)	C-IDDM	0,167	0	1	0,091	N/A
Last.fm (Time-of-Day)	T-IDDM	0,419	0,083	0,9	0,273	1

In comparison, centroid-only approaches, which represent user interest using a single mean embedding per time window, are limited in their ability to detect nuanced changes. While centroid shifts can capture large, abrupt preference transitions, they are insensitive to changes in dispersion, multimodality, or internal reorganization of semantic content. As a result, centroid-based detectors may fail to identify gradual drift or changes that primarily affect the structure of the distribution rather than its mean, as numerically represented in Table 8.1.

Similarly, cosine-similarity–based baselines, which compare aggregated embeddings across time using fixed similarity thresholds, rely on heuristic decision rules that lack statistical grounding. Such approaches are sensitive to embedding noise and scale effects and often require dataset-specific threshold tuning.

In the experiments, cosine-based methods tend to produce unstable detection behavior under sparse or context-confounded conditions, particularly when semantic overlap between time windows remains high despite underlying distributional change. By contrast, the distribution-based formulation employed in C-IDDM evaluates change at the level of empirical distributions and grounds detection decisions in formal

hypothesis testing. This enables sensitivity to a broader class of drift phenomena while maintaining robustness and interpretability.

Table 8.2 Extended Method Comparisons.

Method	Strengths	Limitations	Empirical Behavior
Centroid-only modeling	Computationally efficient; captures large mean shifts	Insensitive to changes in dispersion or multimodality; weak for gradual drift	Misses gradual drift; delayed or absent alarms
Cosine similarity (threshold-based)	Simple to implement; interpretable similarity scores	Heuristic thresholds; noise-sensitive; lacks statistical guarantees	Unstable alarms; elevated false positives under sparsity
Distribution-based (C-IDDM)	Captures mean and structural change; statistically grounded	Higher computational cost per window	Stable detection; low false alarms; consistent delay

These comparisons indicate that the advantages of C-IDDM are not attributable to a single design choice, but rather to the combination of distributional representation, statistical inference, and context-aware segmentation. While point-estimate and similarity-based methods may be suitable for settings with dense data and abrupt changes, as the extended comparison of them has been revealed in Table 8.2, the benchmark results suggest that they are less reliable in the presence of gradual drift, contextual heterogeneity, and sparse interaction streams. The benchmark results, therefore, indicate that distribution-based, context-aware modeling provides a more general and reliable foundation for user interest drift detection than point-estimate or similarity-threshold approaches.

8.4 Role of Temporal Evidence Accumulation

The empirical results indicate that temporal evidence accumulation plays a central role in balancing detection sensitivity and stability within the proposed C-IDDM framework. By integrating statistical evidence over time through EWMA smoothing and p-value fusion, the method emphasizes sustained distributional change while reducing sensitivity to short-term variability in user interaction streams. This mechanism enables drift detection to be driven by accumulated evidence rather than by isolated statistical deviations.

Sensitivity analysis shows that the behavior induced by temporal accumulation is smooth and predictable across a range of parameter settings. Variations in the smoothing strength lead to gradual changes in detection delay and false alarm behavior, rather than abrupt transitions or unstable regimes. This contrasts with single-window or threshold-based detectors, where small parameter changes often produce disproportionate effects on detection outcomes. The results further suggest that accumulation-based strategies are particularly well-suited to user-centric drift detection tasks. In real-world interaction streams, preference evolution often unfolds incrementally and is interleaved with noise arising from irregular activity patterns and contextual shifts. Temporal evidence accumulation allows such gradual changes to be detected without overreacting to transient fluctuations, supporting the design principles outlined in Chapter 4.

8.5 Comparison with Classical Drift Detectors

The comparative evaluation highlights several limitations of classical drift detection methods, such as ADWIN, Page–Hinkley, and KSWIN, when they are applied to semantic, user-level interaction streams. These methods were originally developed for numeric data streams with relatively low dimensionality and stable sampling rates, and they operate under assumptions that are often violated in user-centric textual data, and detailed in Table 7. In particular, they rely on summary statistics or univariate projections that are ill-suited to capturing complex distributional change in high-dimensional embedding spaces.

In the experiments, classical detectors exhibit unstable behavior when applied to sparse and irregular user interaction streams. Short-term fluctuations in activity volume or embedding noise frequently trigger alarms, while gradual semantic drift remains difficult to detect. This behavior is especially pronounced at the individual-user level, where interaction histories are limited and non-stationarity arises from both preference evolution and contextual variation. As a result, these detectors tend to either overreact to noise or fail to respond to meaningful but incremental change. By contrast, C-IDDM is explicitly designed to operate on semantic representations of user behavior and to accommodate the characteristics of user-centric data. Its distribution-based modeling captures changes in the structure of embedding distributions rather than relying on pointwise statistics, and its use of permutation-based testing provides statistically

grounded decisions without strong distributional assumptions. In addition, context-aware segmentation allows drift detection to be performed under homogeneous conditions, reducing the likelihood of confounding effects that commonly affect context-agnostic methods.

The empirical results show that these design choices lead to more stable and interpretable detection behavior compared to classical detectors. Drift signals produced by C-IDDM tend to persist across consecutive time bins and align with sustained semantic change, whereas alarms from classical methods are often isolated and difficult to interpret in terms of user interest evolution. This contrast underscores the need for drift detection approaches that are tailored to the structure and constraints of user-centric, high-dimensional data streams, rather than the direct application of general-purpose detectors developed for system-level or numeric monitoring tasks.

Classical drift detectors are designed under assumptions that align with system-level monitoring tasks, such as tracking numeric performance metrics or sensor readings. When applied to user-centric semantic streams, these assumptions are frequently violated. User interaction data are inherently sparse, context-dependent, and embedded in high-dimensional spaces where drift may manifest as subtle changes in distributional structure rather than as sharp shifts in summary statistics. The mismatch between detector assumptions and data properties explains the unstable or inconclusive behavior observed for classical methods in the empirical evaluation, and motivates the need for drift detection frameworks explicitly designed for semantic, user-level data.

Aspect	Classical Drift Detectors (ADWIN, PH, KSWIN)	User-Centric Semantic Data
Data representation	Univariate or low-dimensional numeric streams	High-dimensional semantic embeddings
Sampling rate	Dense and regular	Sparse and irregular
Drift exhibition	Abrupt changes in mean or variance	Gradual, structural distributional change
Noise characteristics	Limited stochastic noise	High variability due to activity patterns and semantics
Context handling	Implicit or ignored	Explicitly context-dependent

Statistical assumptions	Parametric or window-based summaries	Non-parametric, distribution-free requirements

Table 8.3 Detailed Information of Classical Drift Detectors.

8.6 Insights from Real-World Data

While the Last.fm dataset does not provide explicit ground-truth annotations for user interest drift, the detected drift patterns exhibit qualitative consistency with intuitive changes in listening behavior over time. Detected alarms often coincide with periods of sustained variation in semantic content and with shifts in context-specific listening patterns, suggesting that the proposed method captures meaningful aspects of preference evolution rather than arbitrary fluctuations. Moreover, the variation in detected drift across contexts reflects known differences in user behavior under distinct temporal conditions, lending further plausibility to the observed results.At the same time, the absence of reliable ground truth underscores the inherent challenges associated with evaluating drift detection methods in real-world settings. Without explicit labels, performance assessment must rely on indirect evidence such as temporal coherence, contextual consistency, and alignment with domain knowledge.

In this respect, the real-world evaluation serves primarily as a test of robustness, interpretability, and practical applicability rather than as a definitive measure of detection accuracy. These observations highlight the complementary roles of synthetic and real-world datasets in drift detection research. Synthetic benchmarks provide controlled environments in which detection accuracy, false alarms, and delay can be measured precisely, while real-world data expose methods to the complexity, noise, and heterogeneity characteristic of operational systems. The combined evaluation strategy adopted in this thesis allows the proposed C-IDDM framework to be assessed both under known drift conditions and under realistic deployment scenarios, strengthening the empirical support for its design and applicability.

8.7 Limitations of the Study

Despite the methodological contributions and empirical findings presented in this thesis, several limitations should be acknowledged. First, the proposed framework relies on fixed, pretrained semantic embedding models to represent user interactions. This design assumes representational stability over time, which may not hold in environments where embedding models are frequently updated or retrained. Changes in the embedding space itself could introduce distributional shifts that are unrelated to

genuine user interest drift, potentially confounding detection unless explicitly accounted for.

Second, contextual information is derived from available metadata, such as time-of-day, which provides only a partial characterization of the situational factors influencing user behavior. While temporal context captures meaningful variation in many applications, other contextual dimensions—such as location, device type, social setting, or concurrent activities—are not considered in this study. As a result, some forms of context-dependent preference evolution may remain unobserved or be indirectly reflected in the detected drift signals.

Third, drift detection is performed independently for each user–context pair, without modeling interactions or dependencies at the population level. This design choice emphasizes personalization and interpretability but does not capture collective dynamics, such as emerging trends shared across groups of users. Incorporating population-level information could potentially improve detection power in sparse settings and enable the identification of broader preference shifts.

Finally, while synthetic datasets provide controlled evaluation with known drift characteristics, real-world evaluation is constrained by the absence of explicit ground-truth drift labels. As a result, assessment on real-world data focuses on qualitative consistency, robustness, and comparative behavior rather than definitive accuracy measures. This limitation reflects a broader challenge in drift detection research and motivates the combined use of synthetic and real-world benchmarks. These limitations do not undermine the validity of the proposed approach but instead delineate the scope of the current study. They also suggest several directions for future work, including adaptive embedding alignment, richer context modeling, and hybrid user–population-level drift detection frameworks.

8.8 Implications for Adaptive Personalization Systems

The findings of this thesis have direct implications for the design and operation of adaptive personalization systems. Explicit and statistically grounded drift detection provides a mechanism for distinguishing between stable user behavior and substantive preference evolution, a distinction that is often blurred in systems that rely solely on continuous model adaptation or heuristic decay mechanisms. By treating drift

detection as a separate analytical process, systems can make more informed decisions about when adaptation is warranted and when existing models remain appropriate. In practical deployments, drift signals produced by C-IDDM can be used as triggers for targeted model updates. For example, rather than retraining recommendation models on a fixed schedule or continuously updating parameters, retraining can be initiated only when statistically significant drift is detected for a given user or context. This approach reduces unnecessary computation and mitigates the risk of overfitting to short-term fluctuations.

Context-specific drift detection also enables selective adaptation strategies. When drift is detected in a particular context—such as evening or mobile usage—adaptation can be limited to recommendations or ranking components associated with that context, while preserving stable behavior in others. This selective updating supports finer-grained personalization and avoids global model changes driven by localized preference shifts. Drift signals can further inform exploration and diversification mechanisms within recommender systems. Following a detected drift event, systems may temporarily increase exploration rates or introduce novel content to reassess user preferences under the affected context. Conversely, in the absence of detected drift, systems can maintain exploitative strategies with greater confidence, reinforcing stable preferences.

In monitoring and maintenance workflows, C-IDDM can serve as a diagnostic layer that highlights users or contexts exhibiting frequent or abrupt preference changes. Such information can guide offline analysis, model auditing, or manual intervention, particularly in high-impact personalization scenarios. Because drift alarms are statistically interpretable and context-specific, they support transparent monitoring rather than opaque performance-based alerts. Finally, the modular design of C-IDDM allows it to be integrated alongside existing personalization pipelines without modifying their internal learning objectives. Drift detection can operate asynchronously, producing signals that are consumed by downstream components such as retraining schedulers, feature reweighting modules, or policy controllers. This separation supports scalable deployment and aligns with the broader goal of controlled, interpretable adaptation in user-centric systems.

8.9 Summary

In summary, the experimental results provide consistent evidence supporting the effectiveness of C-IDDM as a context-aware and statistically principled approach to user interest drift detection. Across synthetic and real-world evaluations, the framework demonstrates stable detection behavior, low false alarm rates, and interpretable drift signals, aligning with the theoretical formulation introduced in earlier chapters.

Results on the synthetic datasets illustrate the advantages of context-aware and distribution-based modeling under controlled drift conditions. As shown in the synthetic benchmark visualizations, C-IDDM successfully detects drift events after the designated drift onset while maintaining limited false positives, even in the presence of contextual confounding. In contrast, classical drift detectors and similarity-based baselines exhibit either excessive sensitivity or failure to respond to gradual distributional change, a pattern reflected in the comparative performance dashboards.

The method comparison results further highlight the trade-offs between sensitivity and robustness. Precision–recall and false positive rate comparisons indicate that C-IDDM prioritizes conservative detection, producing fewer spurious alarms than baseline methods while remaining responsive to sustained drift. Heatmap-based analyses reinforce this observation by showing that detected alarms are concentrated in specific user–context combinations rather than being uniformly distributed, supporting the interpretability of context-conditioned detection.

Analysis of alarm timelines and signal strength evolution demonstrates the role of temporal evidence accumulation in stabilizing detection behavior. Drift alarms tend to occur after periods of sustained increase in aggregated evidence rather than in response to isolated fluctuations. Sensitivity analyses across key parameters show smooth and predictable changes in detection behavior, indicating that C-IDDM does not rely on brittle parameter configurations.

On the real-world Last.fm dataset, the detected drift patterns exhibit qualitative consistency with observed changes in listening behavior and context-specific activity. Alarm distributions across users and contexts reveal meaningful heterogeneity, with certain contexts and users contributing more strongly to detected drift. While the

absence of explicit ground truth limits quantitative validation, the coherence of detected signals over time and their alignment with contextual preference structure support the practical relevance of the approach.

These results validate the central design choices of C-IDDM: semantic representation of user behavior, explicit context conditioning, non-parametric distributional testing, and temporal evidence accumulation. At the same time, the discussion acknowledges inherent limitations related to data sparsity, context definition, and real-world evaluation constraints. These findings collectively motivate the concluding remarks and outline clear directions for future research, including richer context modeling, adaptive representations, and hybrid user–user-population-level drift detection. The findings discussed in this chapter highlight both the strengths and the current boundaries of the proposed framework, naturally motivating several avenues for further investigation, which are outlined in the following chapter.

CHAPTER 9
CONCLUSION AND FUTURE WORK

9.1 Conclusion

This thesis explored the problem of user interest drift detection in dynamic, user-centric systems, where preferences evolve over time and unfold with divergent characteristics throughout varying contextual environments. Current methodologies in concept drift detection and personalization were shown to be inadequate for this setting, as they typically rely on labeled outcomes, operate on low-dimensional numeric signals, or treat contextual factors implicitly. Such limitations restrict their ability to capture the nuanced, semantic, and context-dependent nature of real user behavior. In response to these challenges, this thesis introduced the Context-Aware Interest Drift Detection Method (C-IDDM)—a statistically grounded framework explicitly designed to detect changes in users' latent semantic preferences within contextual textual data streams.

Through the act of modeling user interests as context-conditioned semantic distributions and the rigorous conceptualization drift as a distributional change over time, the proposed approach reframes user interest drift detection from an implicit adaptation mechanism into a well-defined statistical decision problem.

C-IDDM integrates several complementary components, including semantic text embeddings, distribution-based comparison, non-parametric statistical hypothesis testing, temporal evidence accumulation, and p-value fusion. The main purpose is to enable drift detection that is robust to noise, interpretable at the user and context level, and applicable in fully unsupervised settings without reliance on predictive error signals or labeled feedback.

The empirical evaluation, supported by extensive visual analysis and benchmark comparisons, demonstrated that C-IDDM consistently outperforms classical drift detectors and context-free baselines on synthetic datasets with known ground-truth drift. In particular, the benchmark figures illustrate that context-aware detection

substantially declines false positive rates while maintaining sensitivity to real preference shifts—an advantage that becomes especially pronounced under confounded and heterogeneous drift scenarios.

On real-world music listening data, where explicit drift labels are unavailable, C-IDDM produced coherent and interpretable drift signals that aligned with intuitive patterns of user behavior and contextual variation. The context-specific alarm distributions, signal strength trajectories, and user–context heatmaps reveal that preference evolution is neither uniform nor temporally isolated, but instead unfolds differently across users and contexts. These findings reinforce the principal motivation of this thesis: aggregating behavior beyond the presence of diverse contexts can mask significant alterations, whereas explicit context conditioning enables the detection of fine-grained and empirically significant drift patterns.

Sensitivity and robustness analyses further confirmed that C-IDDM exhibits stable and predictable behavior encompassing a wide spectrum of parametric configurations, time bin sizes, and data sparsity levels. The empirical evidence suggests that the method avoids brittle threshold effects, maintains low false alarm rates under noise, and degrades gracefully under reduced interaction volume. The present robustness is an immediate outcome of the distributional modeling and temporal evidence accumulation strategy of C-IDDM framework, which favors sustained and statistically supported change over transient fluctuations.

Cumulatively, the results and visualizations represented in this dissertation validate the proposed framework's premise: explicit, context-aware, and statistically principled drift detection is both feasible and advantageous for modeling evolving user interests in modern personalization systems. Beyond methodological contributions, this work provides a practical foundation for deploying drift-aware personalization pipelines, where detected preference shifts can serve as reliable signals for adaptive retraining, recommendation diversification, or user modeling updates. Through the fusion of rigorous statistical techniques and semantic-contextual models, C-IDDM advances the state of the art in user interest drift detection and offers a solid basis for future research in adaptive, interpretable, and user-centric intelligent systems.

9.2 Summary of Contributions

The main contributions of this thesis can be summarized as follows:

1. *Conceptual Contribution*: The thesis structured and introduced the user interest drift as a specialized form of concept drift in user-centric, context-dependent, textual data streams.

2. *Methodological Contribution*: A novel drift detection framework (C-IDDM) is proposed, combining latent semantic representations, distribution-based statistical testing, and temporal evidence accumulation.

3. *Context-Aware Modeling*: The method explicitly incorporated context as a conditioning variable, enabling detection of context-specific interest evolution.

4. *Statistical Grounding*: User interest drift detection is framed as a hypothesis testing problem, providing interpretable and controllable detection decisions.

5. *Empirical Validation*: Large-scale experimental trials on synthetic and real-world datasets demonstrated the effectiveness, robustness, and practical relevance of the proposed approach.

9.3 Implications of Research and Future Work

From a research perspective, this thesis contributed to the literature, bridging the gap between concept drift detection, text mining, and user modeling by proving how statistically grounded change detection could be extended to semantic and contextual domains. The proposed framework demonstrates that drift detection methods are not supposed to be restricted with predictive error monitoring or numeric feature streams, however methods might be formulated directly over high-dimensional textual representations conditioned on context. In this sense, the work provides a novel methodological foundation for future studies on explicit, interpretable drift detection in personalization systems, where understanding *when* and *under what conditions* user interests evolve is as important as adapting to those changes.

From a practical standpoint, C-IDDM offers actionable signals for adaptive recommender systems, content platforms, and user-centric analytics pipelines. Upon the accurate determination of statistically significant preference changes at the user–context level, the framework authorizes more deliberate adaptation strategies, such as

targeted model retraining, context-specific recommendation updates, or controlled exploration. The unambiguous separation between enduring patterns of action and the authentic development of preferences might assist in circumventing the implementation of redundant or premature system upgrades, to improve interpretability of adaptation decisions, and ultimately enhance user trust in personalized services.

While this thesis establishes a comprehensive framework for context-aware user interest drift detection, several promising directions remain open for future research. One natural extension is the integration of adaptive or evolving embedding models, allowing the semantic representation space itself to change over time while preserving the interpretability of detected drift signals. Another direction involves extending the framework to richer, multi-context settings, where multiple contextual factors—such as time, location, device, or social context—interact simultaneously rather than being treated independently.

Future work may also explore population-level drift analysis, aiming to identify collective patterns of interest evolution across groups of users and to study the interaction between individual-level and global drift dynamics. In addition, incorporating downstream adaptation mechanisms—such as automatically triggering recommender retraining, diversification strategies, or exploration policies based on detected drift—represents an instinctive continuation of this work. Finally, broader empirical validation across domains such as news consumption, social media, and e-commerce would further assess the generality and transferability of the proposed approach.

9.4 Final Remarks

As user-centric systems increasingly operate in dynamic, context-rich environments, the ability to understand and detect user interest drift becomes a fundamental requirement rather than an auxiliary concern. This dissertation justifies that drift detection can be formulated as an explicit, statistically grounded problem that directly models semantic and contextual variation in user behavior. The proposed C-IDDM framework demonstrates that incorporating contextual conditioning, distribution-based modeling, and principled statistical testing leads to more reliable and interpretable detection of evolving user interests.

A structured approach to user interest drift detection is established beyond the implicit adaptation and heuristic thresholds, both methodologically sound and practically relevant. The findings suggest that context-aware, statistically principled drift detection can serve as a robust foundation for adaptive personalization systems, enabling more informed system responses and better alignment with long-term user behavior. In this respect, C-IDDM provides a concrete step toward more transparent, adaptive, and user-aligned personalization in real-world applications.

REFERENCES

Aggarwal, C. C. (2007). Data streams: models and algorithms. In *Models and Algorithms*. https://doi.org/10.1007/978-0-387-47534-9

Angelov, P., Lughofer, E. & Zhou, X. (2008). Evolving fuzzy classifiers using different model architectures. *Fuzzy Sets and Systems*, *159*(23), 3160–3182. https://doi.org/10.1016/j.fss.2008.06.019

Azimjonov, J. & Özmen, A. (2021). A real-time vehicle detection and a novel vehicle tracking systems for estimating and monitoring traffic flow on highways. *Advanced Engineering Informatics*, *50*(March). https://doi.org/10.1016/j.aei.2021.101393

Babüroğlu, E. S., Durmuşoğlu, A. & Dereli, T. (2021). Novel hybrid pair recommendations based on a large-scale comparative study of concept drift detection. *Expert Systems with Applications*, *163*, 113786. https://doi.org/10.1016/j.eswa.2020.113786

Baena-Garcia, M., Campo-Avila, J., Fidalgo, R., Bifet, A., Gavalda, R. & Morales-Bueno, R. (2006). Early drift detection method. *4th ECML PKDD International Workshop on Knowledge Discovery from Data Streams*. https://doi.org/10.1.1.61.6101

Barros, Roberto Souto Maior de, Hidalgo, J. I. G. & Cabral, D. R. de L. (2018). Wilcoxon Rank Sum Test Drift Detector. *Neurocomputing*, *275*, 1954–1963. https://doi.org/10.1016/j.neucom.2017.10.051

Barros, Roberto Souto Maior de & Santos, S. G. T. d. C. (2019). An overview and comprehensive comparison of ensembles for concept drift. *Information Fusion*, *52*(December), 213–244. https://doi.org/10.1016/j.inffus.2019.03.006

Barros, Roberto S.M., Cabral, D. R. L., Gonçalves, P. M. & Santos, S. G. T. C. (2017). RDDM: Reactive drift detection method. *Expert Systems with Applications*.

https://doi.org/10.1016/j.eswa.2017.08.023

Barros, Roberto Souto Maior & Santos, S. G. T. C. (2018). A large-scale comparison of concept drift detectors. *Information Sciences*. https://doi.org/10.1016/j.ins.2018.04.014

Basseville, M. & Nikiforov, I. V. (1993). Detection of Abrupt Changes : Mich ` ele Basseville. *Change*, *2*(4), 729–730. https://doi.org/10.1016/0967-0661(94)90196-1

Bifet, A. & Gavaldà, R. (2007). Learning from Time-Changing Data with Adaptive Windowing. In *Proceedings of the 2007 SIAM International Conference on Data Mining*. https://doi.org/10.1137/1.9781611972771.42

Bifet, A., Hammer, B. & Schleif, F. M. (2019). Recent trends in streaming data analysis, concept drift and analysis of dynamic data sets. *ESANN 2019 - Proceedings, 27th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, *August*, 421–430.

Bifet, A., Holmes, G., Kirkby, R. & Pfahringer, B. (2010). MOA: Massive Online Analysis. *Journal of Machine Learning Research*.

Cabral, D. R. de L. & Barros, R. S. M. de. (2018). Concept drift detection based on Fisher's Exact test. *Information Sciences*. https://doi.org/10.1016/j.ins.2018.02.054

Caroprese, L., Pisani, F. S., Veloso, B. M., Konig, M., Manco, G., Hoos, H. & Gama, J. (2025). Modelling Concept Drift in Dynamic Data Streams for Recommender Systems. *ACM Transactions on Recommender Systems*, *3*(2), 1–28. https://doi.org/10.1145/3707693

Cobb, O. & Van Looveren, A. (2022). Context-Aware Drift Detection. *Proceedings of Machine Learning Research*, *162*, 4087–4111.

Elwell, R. & Polikar, R. (2011). Incremental learning of concept drift in nonstationary environments. *IEEE Transactions on Neural Networks*, *22*(10), 1517–1531. https://doi.org/10.1109/TNN.2011.2160459

Fisher, R. A. (1922). On the Interpretation of χ 2 from Contingency Tables, and the Calculation of P. *Journal of the Royal Statistical Society*. https://doi.org/10.2307/2340521

Freund, Y. & Schapire, R. E. (1999). Large margin classification using the perceptron algorithm. *Machine Learning*. https://doi.org/10.1023/A:1007662407062

Frías-Blanco, I., Del Campo-Ávila, J., Ramos-Jiménez, G., Morales-Bueno, R., Ortiz-Díaz, A. & Caballero-Mota, Y. (2015). Online and non-parametric drift detection methods based on Hoeffding's bounds. *IEEE Transactions on Knowledge and Data Engineering*. https://doi.org/10.1109/TKDE.2014.2345382

Gaber, M. M., Zaslavsky, A. & Krishnaswamy, and S. (2011). Mining Data Streams: A Review. *Encyclopedia of Data Warehousing and Mining, Second Edition*. https://doi.org/10.4018/978-1-60566-010-3.ch194

Gama, J., Medas, P., Castillo, G. & Rodrigues, P. (2004). Learning with Drift Detection. *Springer*, 286–295.

Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M. & Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM Computing Surveys*. https://doi.org/10.1145/2523813

Gözüaçık, Ö. & Can, F. (2021). Concept learning using one-class classifiers for implicit drift detection in evolving data streams. *Artificial Intelligence Review*, *54*(5), 3725–3747. https://doi.org/10.1007/s10462-020-09939-x

Han, J., Kamber, M. & Pei, J. (2011). Data Mining: Concepts and Techniques. In *Data Mining: Concepts and Techniques*. https://doi.org/10.1016/C2009-0-61819-5

Hidalgo, J. I. G., Maciel, B. I. F. & Barros, R. S. M. (2019). Experimenting with prequential variations for data stream learning evaluation. *Computational Intelligence*, *35*(4), 670–692. https://doi.org/10.1111/coin.12208

Hinder, F., Vaquet, V. & Hammer, B. (2024). One or two things we know about concept drift—a survey on monitoring in evolving environments. Part B: locating and explaining concept drift. *Frontiers in Artificial Intelligence*, *7*.

https://doi.org/10.3389/frai.2024.1330258

Hoeffding, W. (1963). Probability Inequalities for Sums of Bounded Random Variables. *Journal of the American Statistical Association*. https://doi.org/10.1080/01621459.1963.10500830

Huang, D. T. J., Koh, Y. S., Dobbie, G. & Pears, R. (2015). Detecting Volatility Shift in Data Streams. *Proceedings - IEEE International Conference on Data Mining, ICDM*, *2015-Janua*(January), 863–868. https://doi.org/10.1109/ICDM.2014.50

Hulten, G., Spencer, L. & Domingos, P. (2001). Mining time-changing data streams. *Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *18*, 97–106. https://doi.org/10.1145/502512.502529

Iba, W. & Langley, P. (1992). Induction of One-Level Decision Trees. *International Conference on Machine Learning*. https://doi.org/10.1016/B978-1-55860-247-2.50035-8

Jeong, S. Y. & Kim, Y. K. (2023). Deep Learning-Based Context-Aware Recommender System Considering Change in Preference. *Electronics (Switzerland)*, *12*(10). https://doi.org/10.3390/electronics12102337

Kirkby, R. (1994). *Improving Hoe ff ding Trees by*. *1994*.

Krawczyk, B., Minku, L. L., Gama, J., Stefanowski, J. & Woźniak, M. (2017). Ensemble learning for data stream analysis: A survey. *Information Fusion*, *37*, 132–156. https://doi.org/10.1016/j.inffus.2017.02.004

Krawczyk, B. & Woźniak, M. (2015). One-class classifiers with incremental learning and forgetting for data streams with concept drift. *Soft Computing*, *19*(12), 3387–3400. https://doi.org/10.1007/s00500-014-1492-5

Liu, A., Song, Y., Zhang, G. & Lu, J. (2017). Regional concept drift detection and density synchronized drift adaptation. *IJCAI International Joint Conference on Artificial Intelligence*, *0*, 2280–2286. https://doi.org/10.24963/ijcai.2017/317

McArthur, J. J., Shahbazi, N., Fok, R., Raghubar, C., Bortoluzzi, B. & An, A. (2018). Machine learning and BIM visualization for maintenance issue classification and

enhanced data collection. *Advanced Engineering Informatics*, *38*(October 2017), 101–112. https://doi.org/10.1016/j.aei.2018.06.007

Minku, L. L., White, A. P. & Yao, X. (2010). The impact of diversity on online ensemble learning in the presence of concept drift. *IEEE Transactions on Knowledge and Data Engineering*, *22*(5), 730–742. https://doi.org/10.1109/TKDE.2009.156

Nishida, K. (2008). *Learning and Detecting Concept Drift*. *February*.

Nishida, K. & Yamauchi, K. (2007). Detecting Concept Drift Using Statistical Testing. In *Discovery Science*. https://doi.org/10.1007/978-3-540-75488-6_27

PAGE, E. S. (1954). CONTINUOUS INSPECTION SCHEMES. *Biometrika*. https://doi.org/10.2307/2333009

Pears, R., Sakthithasan, S. & Koh, Y. S. (2014). Detecting concept change in dynamic data streams. *Machine Learning*. https://doi.org/10.1007/s10994-013-5433-9

Pesaranghader, A. & Viktor, H. L. (2016). Fast hoeffding drift detection method for evolving data streams. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9852 LNAI*, 96–111. https://doi.org/10.1007/978-3-319-46227-1_7

Ren, S., Liao, B., Zhu, W. & Li, K. (2018). Knowledge-maximized ensemble algorithm for different types of concept drift. *Information Sciences*, *430–431*, 261–281. https://doi.org/10.1016/j.ins.2017.11.046

Roberts, S. W. (2000). Control Chart Tests Based on Geometric Moving Averages. *Technometrics*. https://doi.org/10.1080/00401706.1959.10489860

Ross, G. J., Adams, N. M., Tasoulis, D. K. & Hand, D. J. (2012). Exponentially weighted moving average charts for detecting concept drift. *Pattern Recognition Letters*, *33*(2), 191–198. https://doi.org/10.1016/j.patrec.2011.08.019

Sakthithasan, S., Pears, R. & Koh, Y. S. (2013). One pass concept change detection for data streams. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*.

https://doi.org/10.1007/978-3-642-37456-2_39

Santos, S. G. T. C., Barros, R. S. M. & Gonçalves, P. M. (2019). A differential evolution based method for tuning concept drift detectors in data streams. *Information Sciences*, *485*, 376–393. https://doi.org/10.1016/j.ins.2019.02.031

Schlimmer, J. C. & Granger, R. H. (1986). Incremental Learning from Noisy Data. *Machine Learning*. https://doi.org/10.1023/A:1022810614389

Serfilippi, L., Bujari, A. & Corradi, A. (2025). Context-aware Drift Detection for Quality-aware Data-Driven Smart City Digital Twins. *GoodIT 2025 - Proceedings of the 2025 International Conference on Information Technology for Social Good*, *January*, 452–459. https://doi.org/10.1145/3748699.3749824

Sethi, T. S. & Kantardzic, M. (2017). On the reliable detection of concept drift from streaming unlabeled data. *Expert Systems with Applications*. https://doi.org/10.1016/j.eswa.2017.04.008

Shaker, A. & Lughofer, E. (2014). Self-adaptive and local strategies for a smooth treatment of drifts in data streams. *Evolving Systems*. https://doi.org/10.1007/s12530-014-9108-y

Silverman, B. W. & Jones, M. C. (1951). E. Fix and J.L. Hodges (1951): An Important Contribution to Nonparametric Discriminant Analysis and Density Estimation: Commentary on Fix and Hodges (1951). *International Statistical Review*, *57*(3), 233–238. https://doi.org/10.2307/1403796

Souto, R., Barros, M. De, Garrido, S. & Santos, T. D. C. (2019). An overview and comprehensive comparison of ensembles for concept drift. *Information Fusion*, *52*(November 2018), 213–244. https://doi.org/10.1016/j.inffus.2019.03.006

Vitter, J. S. (1985). Random sampling with a reservoir. *ACM Transactions on Mathematical Software*, *11*(1), 37–57. https://doi.org/10.1145/3147.3165

Wang, K., Xiong, L., Liu, A., Zhang, G. & Lu, J. (2024). A self-adaptive ensemble for user interest drift learning. *Neurocomputing*, *577*. https://doi.org/10.1016/j.neucom.2024.127308

Wang, Shuo, Minku, L. L. & Yao, X. (2018). A Systematic Study of Online Class Imbalance Learning with Concept Drift. *IEEE Transactions on Neural Networks and Learning Systems*, *29*(10), 4802–4821. https://doi.org/10.1109/TNNLS.2017.2771290

Wang, Sixiang & MacHida, F. (2021). A Robustness Evaluation of Concept Drift Detectors against Unreliable Data Streams. *7th IEEE World Forum on Internet of Things, WF-IoT 2021*, 569–574. https://doi.org/10.1109/WF-IoT51360.2021.9595202

Webb, G. I., Lee, L. K., Goethals, B. & Petitjean, F. (2018). Analyzing concept drift and shift from sample data. *Data Mining and Knowledge Discovery*, *32*(5), 1179–1199. https://doi.org/10.1007/s10618-018-0554-1

Widmer, G. & Kubat, M. (1996). Learning in the presence of concept drift and hidden contexts. *Machine Learning*. https://doi.org/10.1007/BF00116900

Wilcoxon, F. (1945). Individual Comparisons by Ranking Methods. *Biometrics Bulletin*. https://doi.org/10.2307/3001968

Yin, X., Niu, Z., He, Z., Li, Z. & Lee, D. hee. (2020). Ensemble deep learning based semi-supervised soft sensor modeling method and its application on quality prediction for coal preparation process. *Advanced Engineering Informatics*, *46*(June), 101136. https://doi.org/10.1016/j.aei.2020.101136

APPENDIX

C-IDDM Sensitivity Analysis Report

🎯 Executive Summary

Successfully performed comprehensive sensitivity analysis on the dataset, analyzing 45 detection points across 4 contexts and 9 time periods. The analysis reveals important insights about the CIDDM system's performance characteristics.

📊 Dataset Overview

- **Source:** data/lastfm_detection_sample_tod_prepared.csv (958 listening events)
- **Detection Points:** 45 (bin-level aggregates)
- **Users:** 5 users with diverse listening patterns
- **Time Period:** 2023-01-01 - 2023-02-25
- **Contexts:** afternoon, evening, morning, night
- **Parameters:** α=0.05, λ=0.30 (EWMA)

Leakage-Free Protocol

- tuning_fraction: 0.67
- tune_bins: 6
- test_bins: 3
- optimization objective: minimize FPR (primary), minimize delay (secondary)
- constraint: PR-AUC ≥ baseline (T-IDDM) = 0.8816
- delay tolerance w: 1

Best Hyperparameters (selected on tuning bins)

- alpha: 0.05
- ewma_lambda: 0.3
- ref_window_bins: 2
- min_bin_events: 5
- unbiased_ref: False

🎯 Overall Performance Metrics

Metric	Value	Interpretation
Accuracy	0.333	**Moderate**
Precision	1.000	**Good**
Recall	0.091	**Poor**
F1-Score	0.167	**Poor**

Metric	Value	Interpretation
Specificity	1.000	**Good**

Confusion Matrix Analysis

	Predicted	
Actual	**No Drift**	**Drift**
No Drift	**4**	**0** **(4 total)**
Drift	**10**	**1** **(11 total)**
	14	**1** **(15 total)**

- True Positives: 1 (correctly detected drifts)
- False Positives: 0 (false alarms)
- True Negatives: 4 (correctly identified no drift)
- False Negatives: 10 (missed drifts)

Performance by Context

1. Context-Specific Analysis

Context	F1-Score	Precision	Recall	Performance
afternoon	**0.400**	**1.000**	**0.250**	
evening	**0.000**	**0.000**	**0.000**	
morning	**0.000**	**0.000**	**0.000**	
night	**0.000**	**0.000**	**0.000**	

2. Key Insights

1. **Afternoon Context:** Best overall performance with highest F1-score (0.400)

2. **Evening Context:** Complete failure - no drift detection (suggests data sparsity or model limitations)

⚙ **Parameter Sensitivity Analysis**

3. **Alpha (α) Sensitivity**

- **Current Value:** 0.05
- **Best F1-Score:** 0.1667 at α=0.05
- **Recommendation:** Current value appears optimal

4. **EWMA Lambda (λ) Sensitivity**

- **Current Value:** 0.30
- **Best F1-Score:** 0.1667 at λ=0.30
- **Recommendation:** Current value appears optimal
-

🔍 **Detailed Findings**

5. **Strengths**

High Precision: 100.0% - system produces reliable alarms

Low False Positive Rate: 0.0% - system avoids false alarms

Good Specificity: 100.0% - system correctly identifies non-drift periods

6. **Weaknesses**

Low Recall: 9.1% - missing many true drift events

Poor Overall Balance: Low F1-score (0.167) indicates system needs tuning

7. **Critical Issues**

1. **High Miss Rate:** 10 missed drifts vs 1 detected (90.9% miss rate)

💡 **Recommendations**

8. **Immediate Actions**

Lower Alpha Threshold: Consider reducing from 0.05 to 0.01-0.03 to improve recall

Context-Specific Tuning: Implement different thresholds for different contexts based on their performance

9. **System Improvements**

Ensemble Methods: Combine multiple detection approaches for robustness

Context Weighting: Weight different contexts based on their reliability

Temporal Smoothing: Implement temporal smoothing to reduce false alarms

10. Parameter Optimization

Grid Search: Perform systematic parameter search across α and λ

Context-Specific Parameters: Use different parameters for different contexts

Adaptive Thresholds: Implement dynamic threshold adjustment based on data characteristics

📊 Performance Comparison

11. By Time Period

Early Period (bins 3): 9 drift events detected

Middle Period (bins 3): 13 drift events detected

Late Period (bins 3): 11 drift events detected

12. By User Behavior

- **User u1:** 8 drift events detected
- **User u2:** 7 drift events detected
- **User u3:** 4 drift events detected
- **User u4:** 6 drift events detected
- **User u5:** 8 drift events detected

🎯 Success Metrics

13. Achieved

- ✅ **Comprehensive C-IDDM sensitivity analysis**
- ✅ **Parameter sensitivity evaluation across α and λ grid**
- ✅ **Detailed performance breakdown by context**
- ✅ **Actionable recommendations for improvement**

14. Areas for Improvement

- 🔄 **Better recall (target: >0.7)**
- 🔄 **Context balance (target: F1 >0.3 for all contexts)**

🎵 Comprehensive User Preference Insights

Top Artists by Context: Radiohead in afternoon, Metallica in evening, Queen in morning, Bruno Mars in night.

afternoon: Radiohead (20), Dua Lipa (18), Led Zeppelin (14) (out of 29 unique artists)

evening: Metallica (29), Beyoncé (27), Bruno Mars (25) (out of 30 unique artists)

morning: Queen (14), Led Zeppelin (11), Radiohead (11) (out of 25 unique artists)

night: Bruno Mars (15), Travis Scott (14), Beyoncé (14) (out of 26 unique artists)

Listening Activity Distribution: afternoon: 183 events (19.1%), evening: 506 events (52.8%), morning: 93 events (9.7%), night: 176 events (18.4%).

- **Peak Listening Hours: afternoon: 13:00, evening: 21:00, morning: 9:00, night: 22:00.**

Playcount Patterns by Context: afternoon: avg=2.7, median=3.0, evening: avg=2.2, median=2.0, morning: avg=2.3, median=2.0, night: avg=2.0, median=2.0.

User Listening Diversity: Average 15.2 unique artists per user (range: 5-30).

Average Artist Diversity by Context: afternoon: 15.7, evening: 11.0, morning: 15.5, night: 11.2 unique artists per user.

Temporal Activity Trend: increasing (+841.7% change from first to last time bin).

Context Purity: 8.9% of user-time bins are 'pure' (single context), 41 bins have mixed contexts.

Drift Alarms by Context: afternoon: 2 (50.0%), evening: 1 (25.0%), morning: 0 (0.0%), night: 1 (25.0%).

User Segmentation: 2 high-activity users (top 25%), 2 medium, 1 low-activity users.

Top Tags: pop (309), rock (261), hip-hop (149), metal (128), country (28).

🚀 Next Steps

1. **Parameter Tuning:** Implement systematic parameter optimization based on findings
2. **Real Data Testing:** Apply to additional datasets with ground truth
3. **Context Enhancement**: Develop context-specific detection strategies
4. **Production Deployment**: Integrate findings into production system

CURRICULUM VITAE

Surname, Name: BABÜROĞLU, Elif Selen

EDUCATION

PhD, Gaziantep University

Graduate School of Natural and Applied Sciences

Industrial Engineering Department

Title: An Advanced Context-Aware Method for Detecting User Interest Drift: Methodology, Statistical Analysis, And Empirical Evaluation

2019- in progress, Gaziantep/Türkiye

MSc, Gaziantep University

Graduate School of Natural and Applied Sciences

Industrial Engineering Department

Title: A Matched-Pair Comparative Study On Classification Of Data Streams With Concept Drift

2016-2019, Gaziantep/Türkiye

BSc, Gaziantep University

Faculty of Engineering

Industrial Engineering Department

2010-2016, Gaziantep/Türkiye

BSc, Gaziantep University

Food Engineering Department

2013-2016, Gaziantep/Türkiye

PUBLICATIONS

International Journal Publications

Babüroğlu, E.S., Durmuşoğlu, A. & Dereli, T. Concept drift from 1980 to 2020: a comprehensive bibliometric analysis with future research insight. Evolving Systems 15, 789–809 (2024). https://doi.org/10.1007/s12530-023-09503-2

Babüroğlu, E.S., Durmuşoğlu, A. & Dereli, T. Novel hybrid pair recommendations based on a large-scale comparative study of concept drift detection. Expert Systems with Applications 163, (2021). https://doi.org/10.1016/j.eswa.2020.113786

International Conferences

Babüroğlu, E.S., Durmuşoğlu, A. & Dereli, T. Classification of Data Streams with Concept Drift: A Matched-Pair Comparative Study, ICAII4.0 - International Conference on Artificial Intelligence towards Industry 4.0, Nov 15-17, Hatay/Turkey.