1. 서론

메모 포함[준이1]: 고비용 딥러닝보다 피처엔지니어링 +머신 내용을 강화하여 다시 작성하였습니다.

오늘날의 고객은 즉각적으로 구매를 결정하지 않는다. 이메일을 열어보고, 며칠 뒤 모바일 푸시 알림을 확인하며, SMS를 수신한 뒤에야 비로소 결제 버튼을 누른다. 이처럼 산발적으로 흩어진 접점(touchpoints)이 누적되어 하나의 복잡한 고객 여정(Customer Journey)이 완성된다. 기업 입장에서 이 파편화된 기록(log)을 연결하여 누가 구매할지 예측하고, 한정된 마케팅 예산을 어디에 투입할지 결정하는 것은 수익성과 직결되는 지속 가능성을 위한 핵심 과제다. 그럼에도 불구하고 많은 기업 실무에서는 여전히 마지막 접점에만 기여도를 부여하는 Last-Touch 방식이나, 임의 가중치를 부여하는 단순 휴리스틱 모델에 의존하고 있어, 최종 구매에 이르기까지 고객이 거쳐온 선행 접점들의 기여도를 과소평가하고, 구매 의사결정의 인과관계를 지나치게 단순화하는 결과를 초래한다.

이러한 한계를 극복하기 위해 최근 학계와 산업계에서는 마케팅 로그를 시계열 데이터로 간주하고, 순환신경망(RNN)이나 LSTM과 같은 딥러닝 모델을 도입하려는 시도가 활발하다. 일부 선행연구는 딥러닝이 기존 머신러닝(Random Forest, XGBoost 등)을 더 좋다고 보고하며, "주요 해결책은 딥러닝"이라는 암묵적인 전제를 강화해 왔다. 하지만 여기에는 중요한 맹점이 존재한다. 딥러닝의 압도적 성과는 대개 데이터가 풍부하고 품질이 정제된 실험적 환경에서 도출된 결과다. 반면, 실제 기업의 CRM 데이터는 고객의 구매 행동이 드문드문 발생하는 '희소성'이 높고, 이벤트 간의 시간 간격이 불규칙하며, 온라인과 오프라인 기록이 혼재되어 있다. 이러한 희소한 데이터 환경에서는 수많은 파라미터를 가진 복잡한 딥러닝 구조가 이론만큼의 성능 우위를 발휘하지 못할 가능성이 높다. 데이터의 밀도가 낮은 상황에서 무조건적인 고비용 모델 도입은 비효율적일 수 있다는 것이다.

주목할 점은 마케팅 현장에는 이미 검증된 고객 행동 이론이 존재한다는 사실이다. 고객 가치를 평가하는 RFM/BTYD 이론, 월급날이나 요일 효과를 다루는 시점 역동성(Temporal Dynamics), 반응의 즉시성을 보는 최신성(Recency)과 피로도(Fatigue), 콘텐츠의 신선도(Novelty) 등은 마케터들에게 익숙한 직관이자 지식이다. 그러나 기존 연구들은 이러한 도메인 지식을 단순히 현상을 요약하는 지표로만 썼을 뿐, 머신러닝 모델이 직접 학습할 수 있는 입력 피처로 체계적으로 재구성한 시도는 상대적으로 제한적이었다. 인간의 직관(이론)을 기계의 언어(Feature)로 번역하여 주입한다면, 굳이 연산량이 많은 딥러닝 없이도 높은 예측 성능을 달성할 수 있지 않을까?

따라서 본 연구는 "어떤 모델이 가장 고도화되고 복잡한가?"를 묻지 않는다. 대신 "예측 정확도와 비용(구축 및 운영 리소스)을 함께 고려할 때, 어떤 조합이 가장 효율적인가?"라는 실용적 질문에 집중한다. 이를 검증하기 위해 본 연구는 다중 접점 로그 데이터를 기반으로 (1) 장기 가치, (2) 시점 역동성, (3) 행동 최신성/피로도, (4) 콘텐츠 신선도/경로라는 4가지 이론 축을 독립적인 파생변

수 세트로 설계하였다. 그리고 이를 전통적인 트리 기반 모델(XGBoost, Random Forest)과 다양한 딥러닝 모델(RNN, LSTM, CNN-LSTM 등)에 동일하게 투입하여 비교 분석하였다. 이는 도메인 지식 기반의 피처 엔지니어링이 원시 로그에 딥러닝을 바로 적용하는 방식 대비 어느 정도의 한계 기여를 갖는지 실증하기 위함이다.

본 연구가 규명하고자 하는 핵심 질문은 다음과 같다. **RQ1.** 도메인 지식을 반영한 파생변수는 단순 원시 로그(Raw Log)만 사용했을 때보다 예측 성능을 유의미하게 향상시키는가? **RQ2.** 동일한 변수가 주어진다면, '경량화된' 트리 기반 모델과 '연산량이 많은' 딥러닝 모델 중 실무적 비용 효율성이 더 높은 선택은 무엇인가? **RQ3.** 4가지 도메인 지식 축 중 예측력 개선에 가장 결정적인 역할을 하는 핵심 요인은 무엇인가?

결론적으로 본 연구의 차별적 기여도는 다음과 같다. 첫째, 추상적인 고객 행동 이론을 모델이 즉시 학습 가능한 형태의 피처로 구체화하였다. 둘째, 데이터가 희소한 현실 CRM 환경에서는 "잘 가공된 피처와 결합된 경량 모델"이 "원시 데이터 기반의 고비용 딥러닝"보다 더 경제적이고 효율적인 대안이 될 수 있음을 입증하였다. 셋째, 이를 통해 실무자는 고비용 딥러닝 인프라 없이도, 행동 최신성이나 피로도 변수 등을 활용해 타깃팅 정밀도를 즉각적으로 개선할 수 있을 것으로 기대된다.

본 논문의 구성은 다음과 같다. 제 2장에서는 고객 여정 모델링, RFM/BTYD, Temporal Dynamics, 행동 빈도 및 신선도 관련 선행 연구를 고찰하고 본 연구의 피처 설계 근거를 제시한다. 제 3장에서는 베이스라인 모델 및 4가지 확장 버전을 설계하고, 실험에 사용된 데이터와 전처리 과정을 설명한다. 제 4장에서는 각 모델의 성능을 비교 분석하고, 제 5장에서 연구 결과를 요약하며 실무적 시사점 및 한계점을 논의한다.

메모 포함[K2]: 본문 완료 후 마지막에 재작성

2. 문헌연구

2.1. 전통적 연구

(정성 + 정량 연구 동향)

2.2. 지식 활용 연구

(지식을 변수화 해서 반영하는 연구 동향

2.3. 멀티터치 기반 연구

(알고리즘 기반 고도화 동향)

3. 데이터 및 방법론

3.1. 데이터 수집과 패턴

본 연구는 데이터의 신뢰성과 실무 적용 가능성을 확보하기 위해, Kaggle의 "E-commerce multichannel direct messaging 2021–2023" 데이터셋을 분석 대상으로 선정하였다. 이 데이터는 러시아 소재 중형 이커머스 플랫폼에서 2년간(2021~2023) 발생한 실제 CRM 메시징 로그를 마케팅 솔루션 기업 REES46가 익명화하여 공개한 것이다. 원시 데이터(Raw Data)는 고객이 메시지를 수신하고 반응하는 과정을 담은 (1) 전송 로그(messages), 마케팅 의도를 포함한 (2) 캠페인 정보(campaigns), 그리고 (3) 고객 최초 구매일 및 (4) 공휴일 정보의 4가지 테이블로 구성되며 각 범주에 따른 세부 변수들은 아래 <표 1>에 정리하였고, 이러한 변수들은 3.3절에서 기술할 도메인 지식 기반 파생변수 생성의 기초가 된다.

<표 1>데이터 구성

범주	세부 속성
메시지 전송 및 반응 로그	발송 시각, 채널, 플랫폼, 메시지 유형, 이메일 제공자, 반응 여부(Open, Click, Purchase) 등 개별 접점의 속성
캠페인 메타 정보	캠페인 유형(Bulk/Trigger/Transactional), 주제(Topic), 채널, 진행 기간 등 마케팅 전략 속성.
고객 최초 구매일	고객 식별자 및 최초 구매일(First Purchase Date) 등 고객 관계 속성.
공휴일 정보	공휴일 여부 등 외부 시점 속성.

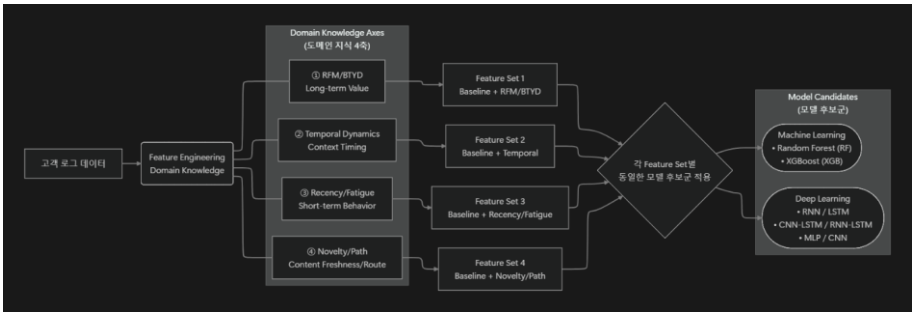
이 데이터가 본 연구의 모델링에 시사하는 구조적 특징은 크게 세 가지로 요약된다. 첫째, 전형적인 '다중 접점(Multi-touch)' 환경을 반영하고 있다. 고객은 단 한 번의 메시지로 즉시 구매하지 않는다. 이메일, 앱 푸시, SMS 등 다양한 채널을 통해 시계열적으로 메시지가 누적되며, 고객이 점진적으로 설득되거나 이탈하는 과정이 데이터에 내재되어 있다. 둘째, 데이터의 '클래스 불균형'이 극심하다. 전체 발송 메시지 중 실제 구매로 이어진 비율은 약 0.12%에 불과하다. 셋째, 정보의 파편화다. 메시지, 캠페인, 고객, 달력 정보가 서로 다른 식별자(Key)로 분리되어 있다. 따라서 이를 통합하여 개별 고객이 시간 흐름에 따라 겪는 경험을 하나의 '시퀀스(Sequence)' 형태로 재구

성하는 전처리 과정이 필수적이다.

3.2. 지식(Knowledge) 기반 파생변수 설계

본 연구는 앞서 2장에서 고찰한 추상적인 마케팅 이론을, 실제 모델이 정량적으로 학습할 수 있는 형태로 변환하는 데 중점을 두었다. 다소 추상적일 수 있는 소비자의 심리를 데이터로 반영하기 위해, (1) 장기 가치(RFM/BTYD), (2) 시점 역동성(Temporal Dynamics), (3) 행동 최신성 및 피로도(Behavior Recency/Fatigue), (4) 콘텐츠 신선도 및 경로(Novelty/Path)라는 4가지 항목을 반영한 파생변수를 설계하였다.

[그림 1] 제안하는 방법론의 프레임워크



[그림 1]은 본 연구가 제안하는 방법론의 전체 프레임워크를 도식화한 것이다. 그림에서 나타난 바와 같이, 정제되지 않은 원시 로그(Raw Logs)는 4가지 도메인 지식 필터를 거쳐 모델 학습에 최적화된 형태의 입력 변수로 재가공 된다.

이는 단순히 데이터를 늘리는 과정이 아니라, 비즈니스 맥락을 데이터에 주입하는 피처 엔지니어링(Feature Engineering)의 핵심 절차이다. 이어지는 제4장 실험에서는 이 변수들을 단순 혼합하는 것이 아니라, 각 지식 축을 개별적으로 모델에 투입하여 각 요인이 예측 성능에 미치는 고유한 기여도를 정밀하게 검증할 예정이다.

아래에서는 베이스라인(Baseline))에 순차적으로 결합되는 네 가지 파생변수 세트의 설계 논리와 계산 방식을 정성적으로 설명한다.

3.2.1. 구매 및 재구매 주기 패턴

메모 포함[준이3]: 밑에 프레임워크 그리부터 변수설계까지 전부 다시 작성하였습니다.

메모 포함[K4]: 그림1을 포함하여 아래 3.2.1과 같은 제목들로 변경해야겠죠?

메모 포함[K5]: 이건 최종 실험 종료후 수정합시다. 전체 과정을 1개의 그림으로 반영해서 "3. 데이터 및 방법론"에 두괄식으로 넣는게 좋을 것 같습니다.

전통적인 마케팅 이론에서 강조하는 '고객 생애 가치 패턴'을 예측 모델에 이식하기 위해 설계되었다. 단순히 마지막 구매 시점을 묻는 것을 넘어, "고객의 개인적인 구매 주기에 비추어 보았을 때, 지금이 다시 구매할 타이밍인가?"를 정량화하는 데 목적이 있다.

첫째, '재구매 준비도'를 통해 고객의 잠재 욕구가 활성화되는 시점을 포착하였다. 고객마다 소비 주기는 천차만별이다. 이를 반영하기 위해 개별 고객의 과거 구매 간격 분포(평균 μ, 변동성 σ)를 정하고, 현재 흐른 시간이 이 평균 주기와 얼마나 일치하는지를 정규분포 형태의 점수로 산출하였다. 이 값이 1에 가깝다면 단순히 시간이 많이 흐른 것이 아니라, 평소 습관대로라면 구매가 임박한 것으로 해석된다. 고객 고유의 '구매 사이클'에 맞춰 적절한 타이밍에 메시지가 도달했음을 알리는 신호다.

$$h_{(t)} = \exp\left(-\frac{1}{2}\left(\frac{\text{days\_since\_last\_purchase} - \mu}{\sigma}\right)^2\right)$$

둘째, '구매 후 불응기'를 정의하여 소비 직후의 만족 효과를 생성하였다. 구매 직후 일정 기간 동안은 추가 마케팅에 대한 반응도가 급격히 떨어지는 현상을 반영한 것이다. 최근 구매 시점으로부터의 경과 시간에 역지수 함수를 적용하여, 구매 직후에는 강한 억제 값(0에 수렴)을 갖고 시간이 지날수록 서서히 억제가 풀리는 구조를 구현하였다. 이는 모델이 방금 구매를 마친 고객에게 불필요한 메시지를 보내는 과적합을 방지하고, 휴식기가 끝나 다시 마케팅을 받아들일 준비가 된 고객을 선별하는 필터 역할을 수행한다.

$$r_{(t)} = \exp\left(-\frac{t}{\tau}\right)$$

3.2.2. 시계열적 메시지 반응 패턴

고객이 메시지에 반응할 확률이 높은 '최적 반응 시점'을 정량화하기 위해 본 연구는 물리적 시간의 흐름 그 자체보다는, Koren(2010)이 제안한 '시간 흐름에 따른 선호 변화' 개념을 확장하여 고객의 생활 패턴과 사회적 맥락이 반영된 시간을 변수화하여 모델에 반영하였다.

첫째, 고객 고유의 '생체 리듬'을 수치화하였다. 일반적인 선형 거리 계산은 밤 23시와 새벽 1시를 멀게 인식하는 오류를 범하기 때문에 이를 방지하기 위해 본 연구는 고객이 과거에 반응(Open/Click/Purchase)했던 시각과 요일 데이터를 24시간 원형 좌표계 위로 매핑하여 고객별 '평균 선호 시간 벡터'를 산출하였고 현재시점과 선호시점 간의 '원형 거리'를 계산하여 0~2 범위로 정규화하였다. 이 값이 0에 수렴할수록, 현재 메시지가 고객이 평소 활동하는 시간대와 정확히 일치하는 것으로 해석된다.

$$d_{norm} = \frac{(1 - \cos(\theta_{curr} - \bar{\theta}))}{2}$$

둘째, '경제적 맥락'을 반영하기 위해 주요 경제 이벤트까지의 근접도를 변수화하였다. 통상적인 급여일(매월 25일)과 월말 정산기 등, 고객의 가처분 소득이 일시적으로 증가하여 '구매 전환 가능성이 높은 시기'를 포착하기 위함이다. 이를 위해 특정 날짜(급여일)에 가까워질수록 값이 1로 급격히 상승하고 멀어지면 0으로 감소하는 정규분포 형태의 '범프 함수'를 적용하였다. 이는 급여 직후 고조되는 소비 심리, '소득 효과'를 모델이 학습하도록 돕는 장치다.

$$bump(t) = \exp\left(-\frac{(t - t_p)^2}{\sigma^2}\right)$$

셋째, 개인을 넘어선 '사회·문화적 시점'을 반영하였다. Koren(2010)의 연구에 따르면 평일과 주말의 사용자 행동 양식은 판이하게 다르다. 따라서 본 연구는 주말(토·일) 여부, 월초의 탐색적 소비와 월말의 목적형 소비 패턴 차이를 반영하는 주차 정보, 그리고 기업의 재고 소진 캠페인이 집중되는 분기 말까지의 잔여일수를 변수항에 추가하였다.

결론적으로 이 변수들은 "언제(Time), 무슨 요일에(Day), "어떤 경제적 상황에 메시지가 도달했는가"을 입체적으로 수치화 한다. 이를 통해 모델은 단순한 텍스트 정보를 넘어, 메시지가 놓인 시공간적 맥락까지 종합적으로 고려하여 반응을 예측할 수 있다.

3.2.3. 마케팅 피로도 패턴

본 피처세트는 고객의 최근 행동 강도와 잦은 마케팅 노출로 인한 피로감을 통합적으로 측정한다. 이는 본 연구의 실험 결과 모델 성능에 가장 높은 기여도를 보인 핵심 축으로, 고객의 즉각적인 반응 성향을 포착하는 데 중점을 둔다.

첫째, '마케팅 피로도' 가설을 검증하기 위해 채널별 노출 강도를 지수 감쇠 형태로 모델링하였다.

메시지 수신 직후에는 피로도가 급증하지만, 시간이 지나면 서서히 해소되는 기억의 망각 과정을 수식으로 구현한 것이다. 이때 피로도가 사라지는 속도인 '시간 상수(τ)'는 선행 연구(Ellis, 2018; Sinch, 2020)의 실증 데이터를 따랐다.

- 이메일(τ =48h): 확인 반응 주기가 길어 피로도가 오래 잔존함.
- 모바일 푸시(τ =24h): 즉각적이나 휘발성이 강해 피로도가 빠르게 해소됨.

- SMS(τ =72h): 침해성이 높아 피로도가 가장 길게 유지됨.

이와 함께 각 채널별 최소 발송 간격 준수 여부를 이진 변수로 추가하여, 모델이 '과잉 마케팅' 여부를 명시적으로 판단하도록 설계하였다.

$$F(t) = \sum_{c \in C} exp\left(-\frac{\Delta t_c(t)}{\tau_c}\right)$$

둘째, 행동의 '최신성'을 통해 반응 가능성을 탐색하였다. 피로도 변수가 '누적된 부담'을 본다면, 최신성 변수는 '마지막 접점'을 본다. 특정 채널로 메시지를 받은 지 얼마나 지났는지, 혹은 채널을 불문하고 마지막 접촉이 언제 였는지를 시간 단위로 계산하였다. 만약 이 값이 매우 크다면 해당 고객은 휴면 상태이며, 현재 메시지가 고객에게 오랜만의 재인식 효과를 줄 수 있는 것으로 판단된다

$$R_{global(t)} = \Delta t_{last\_contact}$$

$$R_{c(t)} = \Delta t_{last\_contact,c}$$

셋째, '슬라이딩 윈도우(Sliding Window)' 기법을 통해 단기 행동 패턴을 집계하였다. 발송 시점을 기준으로 직전 7일과 30일이라는 '이동하는 구간'을 설정하고, 그 안에서 발생한 열람, 클릭, 구매의 빈도와 비율을 계산하였다. 이는 고객의 관심사가 일시적으로 고조된 상태인지, 아니면 지속적으로 유지되고 있는지를 구분하여 단기적인 구매 전환 경향을 포착한다.

넷째, 개별 데이터가 부족한 경우를 대비해 '집단 지성' 정보를 보완재로 활용하였다. 신규 고객이나 데이터가 희소한 고객의 경우, 개인의 이력만으로는 예측이 불가능하다. 이를 해결하기 위해 "현재 메시지의 주제와 채널 조합(예: Sale & Email)"에 대해 전체 고객 집단이 최근 7일/30일간 보인 평균 반응률을 변수로 주입하였다. 이는 개인의 취향을 모를 때 '시장의 최신 트렌드'를 참고하여 예측의 공백을 메우는 역할을 수행한다.

3.2.4. 마케팅 채널 효과성 패턴

고객이 느끼는 콘텐츠의 '진부함'을 측정하고, 현재 마케팅 흐름이 과거의 성공 패턴과 얼마나 닮아있는지를 정량화 하기위해 첫째로, '주제 신선도(Topic Novelty)' 점수를 산출하였다. 동일한 주제(예: "겨울 코트 할인")가 단기간에 반복되면 고객은 피로감을 넘어 무관심해진다. 이를 반영하기 위해 특정 주제의 '최근 7일 내 노출 횟수'와 '경과 시간'을 결합한 감쇠 함수 모델을 적용하였다. 이 점수가 낮다면 해당 주제에 이미 과다 노출되었음으로 판단되며 반대로 점수가 높다면 해당 콘텐츠가 고객에게 신선한 자극으로 다가갈 수 있다고 해석된다.

$$N_{topic(t)} = \exp\left(-\frac{\text{topic\_N7}}{\kappa}\right) \cdot \left(1 - \exp\left(-\frac{\text{topic\_t\_since\_hours}}{\tau}\right)\right)$$

둘째, '경로 일치도(Path Alignment)'를 통해 성공 가능성을 예측하였다. 현재 고객이 겪고 있는 채널 경험의 순서(Sequence)가 과거 구매자들의 '성공 경로'와 얼마나 유사한지를 측정하는 것이다. 본 연구는 실제 구매 발생 직전 5단계의 채널 흐름을 '성공 프로토타입'으로 정의하고, 현재 고객의 경로와의 유사도를 자카드 계수로 산출하였다.

- 예시: 성공 패턴이 Email → Push → App일 때, 현재 고객이 Email → Push 과정을 밟고 있다면 높은 유사도 점수를 받는다. 이는 단순한 빈도 분석을 넘어, 구매를 유발하는 구조적 패턴을 현재 고객이 공유하고 있는지를 판단하는 강력한 지표가 된다.

$$A_{path} = \frac{|P_{current} \cap P_{success}|}{|P_{current} \cup P_{success}|}$$

셋째, '캠페인 속성 재현도'를 분석하였다. 마지막으로 고객이 구매를 하게 했던 바로 그 캠페인의 속성(할인 여부, 이모지 사용, 톤앤매너 등)을 현재 메시지가 얼마나 재현하고 있는지 이진 벡터 유사도로 계산하였다. 값이 1에 가까울수록 과거의 성공 요인을 학습했음을 반영하며, 이는 개인화된 리타겟팅 효율을 높이는 데 기여한다.

$$S_{camp} = \frac{|v_{current} \cap v_{success}|}{|v_{current} \cup v_{success}|}$$

3.3. 데이터 전처리 및 구축

본 연구는 Van Tol(2024)이 제안한 베이스라인 모델의 재현성을 확보하고 학습 효율성을 극대화하기 위해, 선행 연구의 방법론에 근거하여 다음 5단계의 전처리 프로세스를 수행하였다.

1단계: 변수 표준화 및 선행 연구 기반 노이즈 제거

데이터의 일관성을 위해 모든 시간 변수(sent_at 등)를 UTC 기준 Datetime 형식으로 통일하였다. 변수 제거(Feature Selection)에 있어서는 모델의 안정성을 위해 다음 두 가지 기준을 적용하였다.

- 변별력 부재: 전체의 99% 이상이 결측이거나 단일 값인 변수(ab_test, hour_limit 등) 는 예측력이 없으므로 제외하였다.
- 정보의 중복 및 비효율성: subject_length와 같은 단순 텍스트 파생 변수는 차원만 증가시킬 뿐 고객 여정 분석의 핵심 정보가 아니라는 선행 연구의 판단에 따라 배제하고 또한, 정보가 중복되거나 결측률이 높은 기술적 로그 변수들(is_hard_bounced, is_soft_bounced 등)도 함께 제거하였다.

2단계: 화이트리스트 기반 차원 축소

범주형 변수의 차원 저주 문제를 해결하기 위해, 선행 연구에서 검증된 '화이트리스트(Whitelist)' 기준을 준용하였다. 수십만 개의 고유 값을 모두 사용하는 대신, 빈도가 높은 주요 범주만을 선택적으로 유지하는 전략이다.

- 적용: 이에 따라 이메일 도메인은 상위 3개(gmail, mail.ru 등), 플랫폼은 주요 5개 기기로 범주(Desktop, Smartphone, Phablet, Tablet, Other)로 한정하고 나머지는 'Other'로 통합하였다.

3단계: 고객 여정 시퀀스(Sequence) 통합

파편화된 테이블(Messages, Client, Campaigns)을 고객 ID와 캠페인 ID 기준으로 병합(Left Join)하여 단일 시퀀스로 재구성하였다. 특히 공휴일 정보 병합 시에는 분석 데이터가 구매 여부(0 또는 1)에 따라 구조적특성을 고려하여, 오류, 누락 없이 매핑되도록 테이블을 선제적으로 복제하여 결합하였다.

4단계: 시점 제약을 통한 데이터 누수 방지

메시지 발송 시점(t)의 구매 여부를 예측한다는 본 연구의 목적에 맞춰, 입력 변수는 철저히 발송 직전(t-1)까지의 이력만으로 산출하였다. 이는 미래의 정보가 입력되는 '데이터 누수(Data Leakage)'를 방지하기 위한 목적이며, 이를 통해 누적 이력, 최신성, 캠페인 환경 변수 등을 생성하였다. 다섯째, 데이터 불균형 해소를 위한 표본 주입(Infusion) 및 분할(Splitting)을 수행하였다.

5단계: 선행 연구에 따른 표본 주입 및 분할 전체 데이터의 0.12%에 불과한 극심한 클래스 불균형을 해소하기 위해, Van Tol(2024)의 '표본 주입' 방법론을 채택하였다. 원본 데이터(1.7억 건)에서 실제 구매(Positive) 로그를 최대한 추출하여 학습 데이터에 주입함으로써 소수 클래스의 정보 손실을 방지하였다. 이후 비구매(Negative) 데이터를 언더샘플링할 때에는, Thabtah et al.(2020)이 제안한 불균형 데이터의 최적 성능 비율인 약 10%의 전환율을 목표로 설정하여 최종 10만 건의 데이터셋을 구축하였다. 마지막으로, 특정 사용자 집단의 편향을 배제하고 모델의 일반화 성능을 객관적으로 검증하기 위해 구축된 데이터는 인위적 개입 없이 무작위 섞기(Random Shuffle) 후 7:1.5:1.5 비율(Train/Validation/Test)로 분할하였다.

3.4. 학습 및 예측 알고리즘

본 연구는 고객 여정(Customer Journey)의 복합적인 특성인 비선형성, 시계열성, 그리고 국소적 패턴을 다각도로 포착하기 위해, 전통적인 머신러닝 기법부터 최신 하이브리드 딥러닝 아키텍처까지 총 8가지 예측 모델을 구축하여 비교 분석하였다.

3.4.1. 머신러닝 알고리즘: RF, XGB

딥러닝 모델의 성능을 객관적으로 평가하기 위한 기준점(Baseline)으로서, 정형 데이터 분석에 탁월한 트리 기반 앙상블 모델과 기초 신경망을 활용하였다.

- Random Forest (RF) 및 XGBoost (XGB): RF는 배깅(Bagging) 방식을 통해 과적합을 방지하고 안정적인 성능을 제공하며, XGBoost는 부스팅(Boosting) 알고리즘을 기반으로 결측치 처리와 병렬 연산에 강점을 가진다. 이들은 시계열적 순서는 고려하지 않으나, 변수 간의 상호작용과 비선형성을 포착하는 데 효과적이다. 본 연구에서는 도메인 지식 피처(장기 가치(RFM/BTYD)~콘텐츠 신선도 및 경로(Novelty/Path))가 시계열성을 내포하도록 설계되었으므로, 트리 모델이 이러한 피처를 통해 얼마나 성능을 낼 수 있는지 검증하는 척도로 활용된다.

딥러닝 모델 입력 구조: 이어지는 딥러닝 모델(RNN, LSTM, CNN 등)의 효율적인 학습을 위해 입력 데이터를 모델 아키텍처에 적합한 텐서(Tensor) 형태로 재구성하였다. 이때 본 연구는 고도화된 표현 학습 기법보다는, 현업의 원시 로그(Raw Logs) 자체가 가진 정보량의 한계를 객관적으로 규명하는 데 목적을 두었다. 따라서 별도의 임베딩 레이어나 차원 축소 기법을 의도적으로 배제하고, 각 메시지 로그를 하나의 관측치로 보고 수치형 피처만 추출하였다. 결측값은 0으로 대체한 후, 전체 데이터를 $(N_{\text{samples}} \times N_{\text{features}})$형태의 행렬로 구성하고, LSTM 계층 입력을 위해 마지막 축에 채널 차원을 추가하여 $(N_{\text{samples}} \times N_{\text{features}} \times 1)$형태의 3차원 텐서로 변환하였다. 별도의 시퀀스 패딩이나 마스킹은 사용하지 않고, 각 샘플 단위를 직접 모델에 입력하여 원시 데이터의 특성을 보존하였다.

3.4.2. 단일 딥러닝 알고리즘: MLP, CNN, RNN, LSTM

- Multi-Layer Perceptron (MLP): 가장 기본적인 딥러닝 구조로, 입력층과 은닉층, 출력층으로 구성된 전방향 신경망(Feed-forward Neural Network)이다. 고객 행동 변수들의 고차원적 조합을 학습하여 비선형 패턴을 분류하는 데 사용된다.
- Recurrent Neural Network (RNN): 이전 시점의 은닉 상태를 현재 시점의 입력으로 재사용하는 순환 구조를 통해 시계열 데이터를 처리한다. 고객의 최근 행동이 현재의 전환 확률에 미치는 즉각적인 영향을 학습하는 데 적합하다.

- Long Short-Term Memory (LSTM): RNN의 장기 의존성 문제(Long-term Dependency Problem), 시퀀스가 길어질수록 초기 정보가 소실되는 기울기 소실 문제를 해결하기 위해 제안되었다. 입력, 망각, 출력 게이트 구조를 통해 중요한 정보는 장기간 기억하고 불필요한 정보는 삭제한다. 이는 본 연구의 RFM/BTYD(장기 가치(RFM/BTYD))와 같은 장기 가치 신호와 Behavior Recency(행동 최신성 및 피로도(Recency/Fatigue)와 같은 단기 신호를 동시에 처리하는 데 핵심적인 역할을 수행한다.

- Convolutional Neural Network (CNN): 주로 이미지 처리에 사용되나, 시계열 데이터에 1D-CNN을 적용할 경우 시간 축 상의 국소적인 패턴을 효과적으로 추출할 수 있다. 본 연구에서는 특정 기간 내 반복되는 클릭 패턴이나 캠페인 반응의 급등락과 같은 미세 특징을 포착하기 위해 활용되었다.

3.4.3. 하이브리드 딥러닝 알고리즘: CNN-LSTM, RNN-LSTM

단일 모델의 한계를 극복하고 각 구조의 장점을 결합하기 위해, 본 연구는 두 가지 형태의 하이브리드 아키텍처를 제안한다.

- CNN-LSTM: CNN을 인코더로 사용하여 입력 시퀀스에서 노이즈를 제거하고 핵심적인 국소 특징을 추출한 후, 이를 LSTM의 입력으로 전달하여 시계열적 흐름을 학습하는 구조다. 이는 고객의 반응 패턴 중 의미 있는 이벤트를 CNN이 먼저 선별하고, LSTM이 그 이벤트들의 시간적 인과관계를 해석함으로써 예측 정확도를 높이는 전략이다.

- RNN-LSTM: 단순 순환신경망(RNN)과 장기 메모리 신경망(LSTM)을 계층적으로 쌓은(Stacked) 구조이다. 하위 계층의 RNN은 고객의 단기적인 행동 변동성을 민감하게 포착하고, 상위 계층의 LSTM은 이를 바탕으로 장기적인 전환 추세를 안정적으로 학습한다. 이 구조는 본 연구에서 제시한 행동 최신성(Recency)과 피로도(Fatigue)가 복합적으로 작용하는 고객 여정을 모델링하는 데 있어, 단기 반응과 장기 기억을 동시에 최적화할 수 있는 아키텍처로 채택되었다.

이러한 모델들은 4.2절에서 설명할 하이퍼파라미터 최적화 과정을 거쳐 학습 및 평가되었다.

3.5. 예측 성능평가 지표 5종

본 연구는 이항 분류 문제에서 널리 사용되는 다섯 가지 핵심 지표, 정확도(Accuracy), 정밀도(Precision), 재현율(Recall), F1-score, ROC, AUC를 활용하여 모델의 예측 성능을 평가하였다.

정확도(Accuracy)는 전체 관측치 중에서 모델이 올바르게 분류한 비율을 의미하며, 데이터 전반에

대한 직관적인 성능 수준을 제공한다. 그러나 본 연구의 종속변수는 전환 여부(is_purchased)로, 양성 클래스 비율이 약 10%에 불과한 불균형 구조를 가져 모든 관측치를 비전환으로 예측하더라도 높은 Accuracy를 얻을 수 있으므로, 단순 정확도만으로는 모델의 실질적인 변별력을 충분히 판단하기 어렵다.

정밀도(Precision)는 모델이 양성으로 예측한 표본 중 실제로 양성인 비율로, 과대 타깃팅에 따른 비용을 고려할 때 중요하다. 재현율(Recall)은 실제 양성 표본 중 모델이 양성으로 탐지한 비율로, "실제로 전환할 가능성이 있는 고객을 얼마나 놓치지 않고 포착했는가"를 보여주는 지표이다. F1-score는 Precision과 Recall의 조화평균으로, 두 지표 간의 균형을 종합적으로 나타낸다. 마지막으로 ROC AUC는 다양한 결정 임계값(threshold)에 걸쳐 TPR(True Positive Rate)과 FPR(False Positive Rate) 간의 trade-off를 면적 형태로 요약한 것으로, 모델의 전반적인 분류 능력을 임계값에 독립적으로 비교하는 데 유용하다.

따라서 본 연구에서는 불균형 데이터의 특성을 고려하여, 이론적으로는 다섯 지표를 모두 보고하되 해석의 중심을 Recall, F1-score, ROC AUC에 두고 각 모델과 피처 세트의 성능 차이를 논의한다.

4. 연구 결과

메모 포함[준이6]: 베이스라인 비교 별로 유의미한 차이 없음
-> 도메인 지식 적용 피처 엔지니어링의 효과 입증 -> 머신러닝(RF/XGB)이 딥러닝을 이김 흐름으로 재작성 하였습니다

4.1. 파생변수 기초통계 및 상관 구조

본 연구는 앞서 설계한 4가지 도메인 지식 기반 파생변수가 실제 데이터에서 고객 행동의 이질성을 제대로 포착하고 있는지, 그리고 변수 간 정보 중복 없이 독립적인 설명력을 가지는지 검증하기 위해 기초통계량 및 피어슨 상관분석을 수행하였다. 전처리 및 결측 보정이 완료된 최종 데이터셋(N=433,520)의 기술통계량 분석 결과는 <표 2>와 같으며, 주요 특징은 다음과 같다.

첫째, 종속변수(is_purchased)의 평균은 0.100으로 확인되었다. 이는 자연 상태의 극심한 불균형(0.12%)을 그대로 사용하지 않고, 선행 연구(Danny, 2024)의 실험 설계를 준용하여 표본 추출을 통해 학습에 최적화된 클래스 비율로 조정한 결과다.

둘째, 행동 최신성 및 변동성 변수들은 극단적인 '양의 왜도'를 보였다. 데이터의 분포가 왼쪽(0에 가까운 값)으로 치우치고 오른쪽으로 긴 꼬리를 가진 형태다. 특히 feat_last_email_hours(왜도 6.79)와 최근 7일 오픈 횟수인 u_open_cnt_7d(왜도 11.87), 그리고 발송 간격의 표준편차인 u_cadence_std_30d(왜도 87.88)에서 이러한 경향이 두드러졌다. 이는 CRM 데이터 특유의 '롱테일 법칙'이 작용함을 명확히 보여준다. 소수의 '고관여 고객(Heavy User)'이 압도적으로 많은 활동을 수행하는 반면, 대다수 고객은 반응이 없거나 간헐적인 활동만을 보인다는 행동의 양극화가 데이

터에 명확히 반영되어 있다.

셋째, 경로 유사도 변수(feat_path_align)의 평균은 0.0046으로 매우 낮게 나타났다. 이는 마케터가 의도한 이상적인 구매 경로, '이상적 경로'를 그대로 따라가는 고객이 실제로는 극히 드물다는 것을 보여준다. 반면, 주제 신선도(feat_topic_novelty)는 평균과 중앙값이 약 0.41로 유사하게 나타났다. 이는 과거 이력이 전무한 신규 고객이나 비활성 고객의 결측값을 평균적인 수준으로 보정한 결과가 반영되었기 때문이다.

종합하면, 본 데이터는 고객별로 매우 상이한 상호작용 패턴과 높은 희소성을 내포하고 있다. 이러한 통계적 특성은 단순한 선형 모델로는 포착하기 어려운 비선형적 정보를 담고 있으며, 본 연구가 제안한 파생변수들이 이러한 복잡한 행동 패턴을 모델에 효과적으로 전달하는 매개체가 될 것임을 기대한다.

<표 2> 도메인 지식 기반 파생변수의 기초 통계량 요약

그룹 (Group)	변수명 (Variable)	Mean	Std	Min	Median	Max	Skewness
Target	is_purchased	0.1	0.3	0	0	1	2.6667
RFM/BTYD (장기 가치)	feat_rtb_hazard	0.9526	0.1713	0	0.9933	1	-4.4794
	feat_postbuy_refrac	0.9645	0.1768	0	1	1	-4.9744
Temporal(시점 역동성)	feat_hour_shift	0.5673	0.5244	0	0.3831	2	1.2387
	feat_dow_shift	0.874	0.6482	0	0.8996	2	0.2242
	feat_payday_bump	0.1412	0.2564	0	0.0022	1	1.8939
	feat_monthend_bump	0.1389	0.3329	0	0	1	2.1266
Recency (최신성/피도)	feat_fatigue	0.223	0.407	0	0	1.979	1.342
	feat_last_email_hours	455.8823	1439.8008	0	315	17232.49	6.7904
	u_open_cnt_7d	0.0365	0.2679	0	0	11	11.868
	u_buy_rate_30d	0.0179	0.1235	0	0	1	7.2046
	u_cadence_std_30d	2.544	54.9354	0	0	11154.64	87.8849
Novelty(신선도/경로)	feat_topic_novelty	0.4111	0.0579	0	0.4111	0.7553	-1.1563
	feat_path_align	0.0046	0.0668	0	0	1	14.6579
	feat_like_last_success	0.0178	0.1237	0	0	1	7.5261

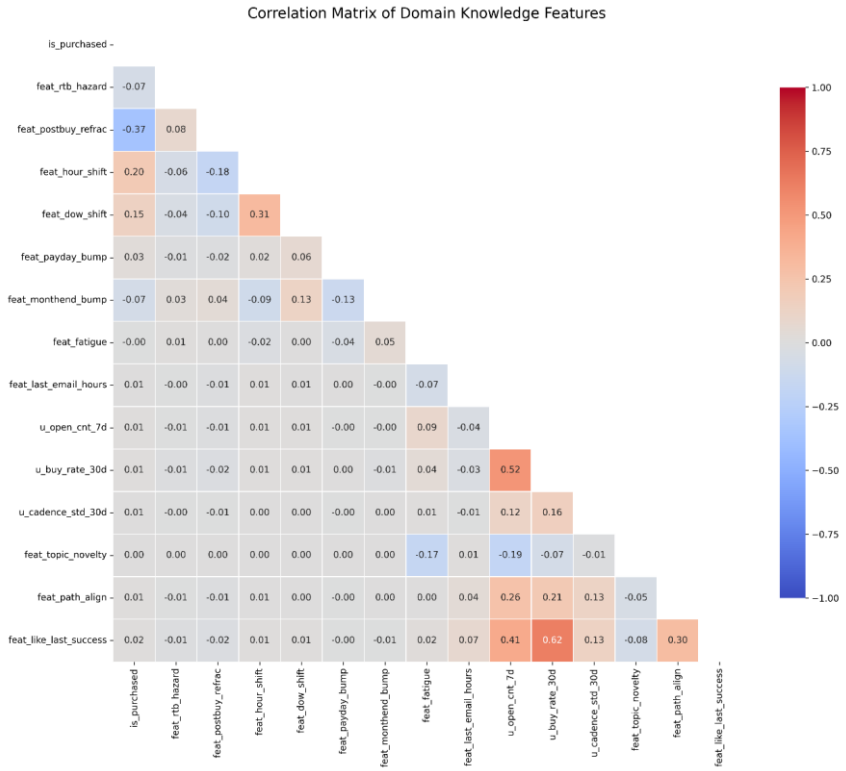
파생변수들이 서로 중복된 정보를 제공하지 않고 독립적인 설명력을 갖는지 검증하기 위해, 피어슨 상관분석을 수행하였다. 분석 결과는 [그림2]에서 확인할 수 있으며, 데이터의 구조적 특성은 다음 두 가지로 요약된다.

첫째, 타겟 변수(is_purchased)와 파생변수 간의 선형 상관관계가 매우 낮게 나타났다($|r| < 0.10$). 이는 고객의 구매 결정이 특정 단일 변수의 증감에 따라 비례적으로 이루어지는 단순한 구조가 아니라고 해석된다. 구매는 하나의 강력한 요인이 아니라, 여러 변수가 복합적으로 작용할 때 발생하는 비선형적(Non-linear) 사건임을 통계적으로 확인하였다

둘째, 설명 변수 간의 상관계수 역시 대부분 0.20 미만($|r| < 0.20$)으로 나타나, 변수들끼리 강한 상관관계를 가져 모델이 변수의 영향력을 중복으로 계산하게 만드는 문제인 '다중공선성'이 발견되지 않았다. 본 연구의 변수들은 이 수치가 낮게 나타남으로써, 각 변수가 '행동의 최신성', '콘텐츠의 신선도' 등 서로 다른 차원의 정보를 독립적으로 제공하고 있음을 입증하였다. 도메인 지식에 기반한 변수 설계가 정보의 중복 없이 효율적으로 이루어졌음을 보여준다.

종합적으로 볼 때, 본 연구의 파생변수들은 타겟 변수와 단순한 선형적 관계를 보이지는 않는다. 그러나 이는 변수의 효용이 낮다는 의미가 아니라, 오히려 인간의 복잡한 구매 행동 패턴을 담고 있기에 비선형적 설명력을 보유하고 있는 것으로 해석된다. 이러한 변수 세트의 결합 효과가 예측 성능에 미치는 실질적 기여도는 제4장 실증 분석을 통해 구체적으로 증명될 것이다.

[그림 2] 주요 파생변수 간 피어슨 상관관계 히트맵



4.2. 실험 설계 및 분석 프레임워크

본 장에서는 제1장에서 제시한 연구 질문 중, 알고리즘 비교(RQ2)와 도메인 지식 기반 피처세트의 효과(RQ1, RQ3)를 순차적으로 검증한다. 먼저 4.3절에서 알고리즘별 성능을 비교하고(RQ2), 4.4절에서 지식 기반 피처세트가 베이스라인 대비 얼마나 성능을 향상시키는지와 각 세트의 한계 기여도를 분석한다(RQ1, RQ3). 실험의 목적은 (1) 선행연구인 Van Tol(2024)에서 보고된 것처럼 RNN-LSTM 계열 모델이 여전히 최적의 선택인지, (2) 제안한 네 가지 마케팅 도메인 지식 축(RFM/BTYD, Temporal Dynamics, Behavior Recency/Fatigue, Novelty/Path)을 입력 피처로 추가할 때 어떤 피처 세트가 예측 성능 향상에 가장 크게 기여하는지를 정량적으로 검증하는 데 있다.

이를 위해 본 연구는 3.3절에서 구축한 약 10만 건 규모의 학습용 표본(테스트 세트 크기 n_eval ≈ 15,000, 양성 비율 pos_rate_eval ≈ 0.10)을 사용하였다. 종속변수는 메시지 발송 후 24시간 이내 구매 여부(is_purchased)이며, 불균형(class imbalance) 구조를 고려하여, 본 연구에서는 Accuracy보

다 Precision, Recall, F1-score, AUC에 분석의 비중을 두었다.

모델 구성 측면에서, 본 연구는 단일 알고리즘에 의존한 결론이 구조적 편향(structural bias)에 의해 왜곡되는 것을 방지하기 위해, 전통적인 머신러닝부터 시계열 딥러닝, 하이브리드 모델까지 총 8개 알고리즘을 비교 대상으로 선정하였다. 구체적으로는 Random Forest(RF), XGBoost(XGB), Multi-Layer Perceptron(MLP), CNN, RNN, LSTM, CNN-LSTM, RNN-LSTM으로 구성하였다. 각 알고리즘은 공통된 학습·검증·테스트 분할을 공유하며, 입력으로는 3.2절에서 설계한 도메인 지식 기반 파생변수와 베이스라인 로그 피처를 사용한다.

<표 3> 비교 실험 구성

실험 구성	추가 변수군 (Added Features)	주요 특징 및 기대 효과
Baseline	없음 (Raw Logs)	마케팅 해석이 배제된 원시 로그 데이터
+RFM/BTYD	장기 가치 변수	고객의 재구매 주기 및 이탈 위험 등 장기 패턴
+Temporal	시점 역동성 변수	요일, 월말, 급여일 등 구매가 일어나는 시간적 맥락
+Recency	행동 최신성/피로도 변수	직전 반응 행동 및 잦은 알림으로 인한 피로도
+Novelty	신선도/경로 변수	콘텐츠의 신규성 및 과거 성공 경로와의 유사도

4.2 하이퍼파라미터 탐색 및 최적화

학습 과정에서 모든 모델은 동일한 학습·검증 프로토콜을 따른다. 우선 3.3절에서 정의한 학습(Train), 검증(Validation), 테스트(Test) 세트의 분할을 유지하되, 학습 단계에서는 5-fold 교차검증(Cross-Validation)을 추가로 적용하였다. 이는 특정 데이터 조합에 의해 우연히 성능이 높게 나타나는 편향을 방지하기 위함이다. 각 알고리즘–버전 조합에 대해 Optuna 기반 Bayesian Optimization을 수행하여 주요 하이퍼파라미터(예: 트리 개수, 은닉층 차원, 드롭아웃 비율, 학습률 등)를 탐색하였다. 이는 무작위로 값을 탐색하는 랜덤 서치보다 효율적으로 최적해를 찾아내는 방식이다. 검증 세트의 ROC AUC를 최대화하는 파라미터 조합을 선택하고, Early Stopping과 Pruning을 적용하여 과적합을 방지하였다.

본 연구에서 사용된 모든 머신러닝·딥러닝 모델(Random Forest, XGBoost, MLP, CNN, RNN, LSTM, CNN-LSTM, RNN-LSTM)은 Optuna 기반 Bayesian Optimization 기법을 사용하여 주요 하이퍼파라미터를 자동 탐색하였다.

탐색 범위(Search Space)는 각 모델의 구조적 특성을 반영하여 개별적으로 설정하였으며, 최종적으로는 검증 세트(Validation Set)의 AUC를 최대화하는 방향으로 최적값(best hyperparameters)을 도출하였다. 표 4는 각 알고리즘별로 탐색한 파라미터 범위와 Optuna가 선택한 최적값을 요약한

메모 포함[K7]: 라이브러리가 저장한 함수를 쓴건지 본인이 직접 계산한걸 쓴건지...

왜 필요하냐면, Precision, Recall, F1-score 그림을 보면서 F1-score가 계산이 잘 된건지 확인 필요 Learning Curve

메모 포함[준이8R7]: 직접계산 AUC로 수정하였습니다

것이다.

<표 4-1. 머신러닝·딥러닝 모델의 하이퍼파라미터 탐색범위 및 최적값 요약>

알고리즘	파라미터	탐색범위	최적값
Random Forest	n_estimators	100 ~ 800 (step=50)	700
	max_depth	10 ~ 30 (step=5)	30
	min_samples_split	2 ~ 10	2
	min_samples_leaf	1 ~ 5	8
XGBoost	n_estimators	100 ~ 1,000	450
	learning_rate	0.001 ~ 0.3 (log)	0.0608
	max_depth	3 ~ 10	3
	min_child_weight	1 ~ 10	4
	subsample	0.5 ~ 1.0	0.5733
	colsample_bytree	0.5 ~ 1.0	0.5461
	gamma	0 ~ 5	0.9313
	reg_alpha	1e-5 ~ 10 (log)	0.0011
	reg_lambda	1e-5 ~ 10 (log)	0.0002
MLP	n_layers	1 ~ 3	2
	activation	relu/gelu/selu	Selu
	units_l0	32 ~ 512	128
	dropout_l0	0.0 ~ 0.5	0.208
	l2	1e-6 ~ 1e-2	4.87E-03
	optimizer	adam/rmsprop/sgd	Adam
	learning_rate	1e-4 ~ 1e-2	2.83E-04
	batch_size	16/32/64/128	32
CNN	conv_filters	32 ~ 256	224
	kernel_size	2 ~ 5	2
	pool_size	2 ~ 3	3
	num_conv_layers	1 ~ 2	2
	dense_units	32 ~ 256	192
	dropout_rate	0.1 ~ 0.6	0.24
	l2	1e-6 ~ 1e-2	1.44E-04
	learning_rate	1e-5 ~ 1e-3	1.17E-04
	batch_size	32/64/128	32
RNN	num_layers	1 ~ 2	2
	rnn_units1	32 ~ 256	224
	rnn_units2	16 ~ 128	128

	dense_units	16 ~ 256	96
	dropout_rate	0.0 ~ 0.5	0.3461
	l2	1e-6 ~ 1e-2	3.20E-03
	activation	tanh/relu/sigmoid	tanh
	learning_rate	1e-5 ~ 1e-2	1.21E-05
	batch_size	32/64/128	32
LSTM	num_lstm_layers	1 ~ 2	2
	lstm_units_1	32 ~ 256	96
	lstm_units_2	32 ~ 256	112
	dense_units	16 ~ 64	48
	dropout_rate	0.1 ~ 0.5	0.453
	l2	1e-5 ~ 1e-2	7.43E-04
	learning_rate	1e-5 ~ 1e-3	3.18E-04
	batch_size	32/64/128	128
CNN-LSTM	conv_filters	16 ~ 128	48
	kernel_size	2 ~ 5	4
	pool_size	2 ~ 4	4
	lstm_units	32 ~ 256	256
	dense_units	16 ~ 64	16
	dropout_rate	0.1 ~ 0.5	0.1156
	l2	1e-5 ~ 1e-2	3.23E-05
	learning_rate	1e-5 ~ 1e-3	5.71E-04
	batch_size	32/64/128	128
RNN-LSTM	lstm_units1	32 ~ 256	256
	lstm_units2	16 ~ 128	64
	dense_units	16 ~ 64	64
	dropout_rate	0.1 ~ 0.5	0.4873
	l2	1e-5 ~ 1e-2	7.33E-04
	learning_rate	1e-5 ~ 1e-3	1.77E-04
	batch_size	32/64/128	32

.

4.3. 알고리즘별 기초 예측 성능 비교

메모 포함[K9]: 필요없을 듯한데 일단은 삭제하지 말고 두시오

먼저, 마케팅 차원의 전문적 해석을 배제하고, 사용자 로그 그 자체인 원시 로그(Raw Logs)만을 투입 변수로 설정하여 베이스라인 성능을 진단하였다(<표 4-2> 참조)..

분석 결과, 통상적으로 시계열 처리에 강점이 있다고 알려진 딥러닝 모델(RNN, LSTM 계열)이 전통적인 머신러닝 방법론(Random Forest, XGBoost) 대비 뚜렷한 성능 우위를 보이지 못하였다. 구

체적인 수치인 F1-score를 기준으로 보면, Random Forest(0.7077)가 딥러닝 기반인 LSTM(0.6981)보다 오히려 소폭 앞서는 경향을 보였다. 물론 RNN-LSTM 결합 모델이 0.7106으로 가장 높은 수치를 기록했으나, Random Forest와의 격차는 0.0029(약 0.4%)에 불과하다. 이는 통계적 관점이나 비용을 고려해야 하는 실무적 관점에서 볼 때, 유의미한 성능 차이라 정의하기 어렵다.

이러한 현상은 이커머스 데이터가 가진 태생적 한계인 '데이터 희소성'과 '불규칙한 이벤트' 때문인 것으로 풀이된다. 고객은 매일 규칙적으로 접속하지 않고 드문드문 방문하며, 행동 간의 시간차 또한 일정하지 않다. 딥러닝이 데이터 사이의 맥락을 스스로 학습하기에는 정보의 공백이 너무 큰 셈이다.

따라서, 데이터의 특성을 고려하지 않은 채 단순히 모델의 복잡도만 높이는 것은 예측 성능 향상을 담보하지 못한다. 오히려 도메인 지식을 활용해 듬성듬성한 데이터 사이의 맥락을 이어주는 특징 추출(Feature Extraction) 과정이 선행되어야만 모델이 유의미한 패턴을 학습할 수 있을 것으로 판단된다..

<표 4-2 알고리즘별 베이스라인 적용 성능 비교>

Algorithm	Accuracy	Recall	F1-score	AUC
Name	Baseline	Baseline	Baseline	Baseline
RF	0.9493	0.6013	0.7077	0.9665
XGB	0.9497	0.5954	0.707	0.965
MLP	0.9449	0.5288	0.662	0.9606
RNN	0.9487	0.5739	0.6952	0.9646
CNN	0.9373	0.4529	0.5956	0.9417
LSTM	0.9479	0.5908	0.6981	0.9641
CNN-LSTM	0.9494	0.602	0.7082	0.9652
RNN-LSTM	0.9491	0.6131	0.7106	0.9648

4.3. 지식 기반 타겟 마케팅 예측성능

본 연구에서 설계한 4가지 도메인 지식 축(장기 가치, 시점 역동성, 행동 최신성/피로도, 콘텐츠 신선도/경로)을 단계적으로 반영하는 실험을 수행하였다. 이를 통해 각 변수 그룹이 모델의 예측력을 얼마나 끌어올리는지, 모델 성능 개선에 미치는 영향을 독립적으로 평가하였다.

4.3.1. 행동 최신성 및 피로도(Recency/Fatigue)의 정보 가치

실험 결과, 네 가지 지식 축 중 '행동 최신성 및 피로도(Recency/Fatigue)' 변수군이 타 변수 대비 압도적인 성능 향상을 이끌어내는 핵심 요인임이 확인되었다.

메모 포함[K10]: 4.3.1~4.3.2로 반영하는 것도 좋은데, 독자들한테는 가독성이 좋지 않을 것 같아요. 두괄식으로 주장하려고 한건 좋은데..

3장에서 설계한 흐름과 맞추는게 좋습니다.
4.3.1. 구매 및 재구매 주기 패턴 성능
4.3.2. 시계열적 메시지 반응 패턴 성능
4.3.3. 마케팅 피로도 패턴 성능
4.3.4. 마케팅 채널 효과성 패턴 성능

또한 지금 버전의 4.3.1은 4.5 내용과 방향이 같아서 합하는게 나을 듯 합니다.

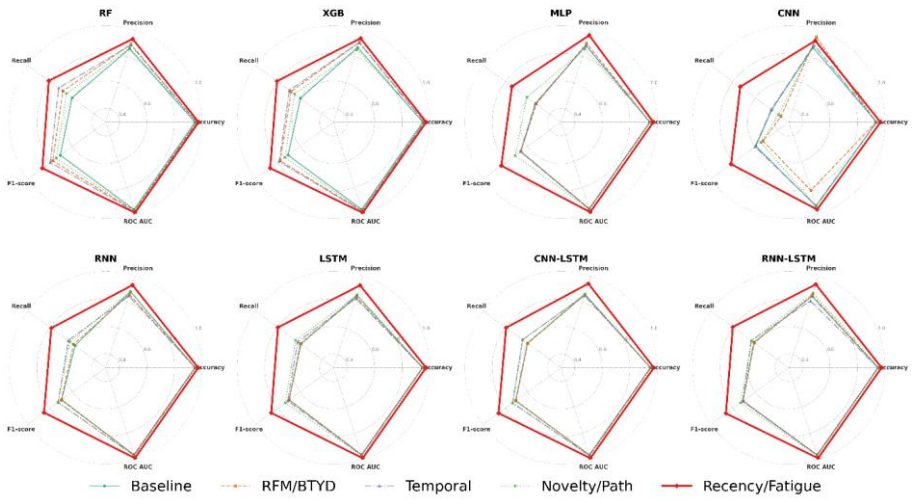
<표 4-3>과 같이 해당 변수 세트를 적용했을 때 모델들의 평균 성능은 베이스라인 대비 비약적으로 상승하였는데, 특히 Random Forest(RF) 모델의 변화가 가장 극적이다. F1-score는 베이스라인 0.7077에서 0.8683으로, 실제 구매자를 찾아내는 능력인 재현율(Recall)은 0.6013에서 0.8105로 현저히 향상하였다.

이는 고객의 구매 의사를 예측함에 있어, '가입한 지 얼마나 되었는가'와 같은 과거의 장기 이력이나 정적인 속성보다, '지금 당장 고객이 지쳐있는가, 아니면 반응하고 있는가(직전 반응 행동)'가 훨씬 강력한 설명력을 가진다는 사실을 실증한다.

구체적으로 살펴보면, '최근 7일간의 앱 실행 횟수(u_open_cnt_7d)'와 같은 단기 활동성 지표가 모델에 투입됨으로써, 딥러닝이 데이터의 불연속성 때문에 학습하기 어려웠던 '잠재적 구매 의도'를, 도메인 지식 피처가 모델이 인지 가능한 형태의 신호로 변환하여 전달한 효과로 판단된다.

[그림 4-1]의 시각화 자료에서 볼 수 있듯, 해당 변수가 추가되는 순간 모델의 성능 지표가 그래프 외곽으로 크게 확장되는 양상을 확인할 수 있다.

[그림 4-1] 알고리즘별 성능 변화 레이더 차트 비교 (Performance Evolution by Model)



<표 4-3> 행동 최신성 및 피로도(Recency/Fatigue) 피처 적용에 따른 성능 변화

Algorithm	Accuracy	Recall	F1-score	AUC

Name	Baseline	Recency/Fatigue	Baseline	Recency/Fatigue	Baseline	Recency/Fatigue	Baseline	Recency/Fatigue
RF	0.9493	**0.9749**	0.6013	**0.8105**	0.7077	**0.8683**	0.9665	**0.991**
XGB	0.9497	**0.9749**	0.5954	**0.8059**	0.707	**0.8677**	0.965	**0.9918**
RNN-LSTM	0.9491	**0.9743**	0.6131	**0.8026**	0.7106	**0.8642**	0.9648	**0.9908**
CNN-LSTM	0.9494	**0.9739**	0.602	**0.7935**	0.7082	**0.8613**	0.9652	**0.9909**
LSTM	0.9479	**0.9731**	0.5908	**0.7974**	0.6981	**0.8579**	0.9641	**0.9904**
RNN	0.9487	**0.9724**	0.5739	**0.7882**	0.6952	**0.8535**	0.9646	**0.99**
MLP	0.9449	**0.9706**	0.5288	**0.7405**	0.662	**0.8371**	0.9606	**0.9868**
CNN	0.9373	**0.9667**	0.4529	**0.7359**	0.5956	**0.8183**	0.9417	**0.9669**
Average	0.9470	**0.9726**	0.5698	**0.7843**	0.6856	**0.8535**	0.9616	**0.9873**
Gain	**2.70%**		**37.65%**		**24.50%**		**2.68%**	

4.4.2. 기타 도메인 지식(장기 가치, 시점, 신선도)의 제한적 기여

반면, 앞서 검증한 '행동 최신성'을 제외한 나머지 도메인 지식 변수들은 상대적으로 제한적인 성능 개선 효과를 보였다(<표 4-4>~<표 4-6> 참조).

구체적인 수치를 살펴보면, 요일이나 시간대 정보를 담은 시점 역동성(Temporal) 변수는 F1-score 기준 평균 4.57%의 향상을 이끌어내며 차 순위 기여도를 기록했으나, 최신성 변수의 타 모델 대비 높은 예측력에는 미치지 못했다. 또한, 고객의 등급이나 충성도를 대변하는 장기 가치(RFM/BTYD)와 콘텐츠 신선도(Novelty/Path) 변수는 각각 평균 1.36%, 3.45%라는 미미한 상승폭을 기록하는 데 그쳤다.

이는 본 연구가 다루는 이커머스 마케팅 환경의 특수성에 기인한다. 고객은 '내가 이 쇼핑몰의 오랜 우수 고객이므로 구매한다'는 식의 장기적·계획적 로열티보다는, '지금 이 순간 알림이 와서' 혹은 '심심하던 차에 눈에 띄어서'와 같은 즉각적인 자극에 반응하는 경향이 강하다. 소비 행태가 다분히 '충동적'이고 '맥락 의존적'임을 뒷받침하는 결과라 할 수 있다.

따라서 실무적 관점에서 볼 때, 수년 치의 방대한 과거 이력을 모두 가공하려 리소스를 투입하는 것은 비용 대비 효과가 낮다. 오히려 최근 행동 데이터를 정교하게 변수화하는 데 집중하는 것이 시스템 효율성과 예측 성능을 동시에 확보하는 최적의 전략임이 확인되었다.

<표 4-4> 시점 역동성(Temporal) 피처 적용에 따른 모델별 성능

Algorithm	Accuracy		Recall		F1-score		AUC	
Name	Baseline	Temporal	Baseline	Temporal	Baseline	Temporal	Baseline	Temporal

RF	0.9493	**0.9624**	0.6013	**0.7222**	0.7077	**0.7967**	0.9665	**0.9834**
XGB	0.9497	**0.9615**	0.5954	**0.6948**	0.707	**0.7865**	0.965	**0.9812**
CNN-LSTM	0.9494	**0.9522**	0.602	**0.6458**	0.7082	**0.7338**	0.9652	**0.9707**
RNN	0.9487	**0.9512**	0.5739	**0.6359**	0.6952	**0.7267**	0.9646	**0.9697**
RNN-LSTM	**0.9491**	0.9477	0.6131	**0.6373**	0.7106	**0.7132**	0.9648	**0.9682**
LSTM	0.9479	**0.9483**	0.5908	**0.6203**	0.6981	**0.7098**	0.9641	**0.9673**
MLP	0.9449	**0.9452**	0.5288	**0.532**	0.662	**0.6645**	0.9606	**0.9646**
CNN	0.9373	**0.9388**	0.4529	**0.4569**	0.5956	**0.6036**	**0.9417**	0.9362
Average	0.9470	**0.9509**	0.5698	**0.6182**	0.6856	**0.7169**	0.9616	**0.9677**
Gain	**0.41%**		**8.49%**		**4.57%**		**0.63%**	

<표 4-5> 장기 가치(RFM/BTYD) 피처 적용에 따른 모델별 성능

Algorithm	**Accuracy**		**Recall**		**F1-score**		**AUC**	
Name	**Baseline**	**RFM/BTYD**	**Baseline**	**RFM/BTYD**	**Baseline**	**RFM/BTYD**	**Baseline**	**RFM/BTYD**
XGB	0.9497	**0.9604**	0.5954	**0.6784**	0.707	**0.7775**	0.965	**0.9759**
RF	0.9493	**0.9596**	0.6013	**0.6843**	0.7077	**0.7756**	0.9665	**0.9752**
RNN-LSTM	0.9491	**0.9495**	**0.6131**	0.6098	0.7106	**0.7111**	0.9648	**0.9656**
CNN-LSTM	**0.9494**	0.9489	**0.602**	0.6	**0.7082**	0.7056	0.9652	**0.9674**
RNN	**0.9487**	0.9481	0.5739	**0.5882**	0.6952	**0.6982**	0.9646	**0.9648**
LSTM	**0.9479**	0.9471	0.5908	**0.5961**	**0.6981**	0.6967	0.9641	**0.9643**
MLP	0.9449	**0.9455**	**0.5288**	0.5229	**0.662**	0.6617	**0.9606**	0.9602
CNN	**0.9373**	0.9337	**0.4529**	0.3699	**0.5956**	0.5325	**0.9417**	0.8211
Average	0.9470	**0.9491**	0.5698	**0.5812**	0.6856	**0.6949**	**0.9616**	0.9493
Gain	**0.22%**		**2.01%**		**1.36%**		**-1.27%**	

<표 4-6> 콘텐츠 신선도(Novelty/Path) 피처 적용에 따른 모델별 성능

Algorithm	**Accuracy**		**Recall**		**F1-score**		**AUC**	
Name	**Baseline**	**Novelty/Path**	**Baseline**	**Novelty/Path**	**Baseline**	**Novelty/Path**	**Baseline**	**Novelty/Path**
RF	0.9493	**0.9542**	0.6013	**0.6516**	0.7077	**0.7438**	0.9665	**0.9743**
CNN-LSTM	0.9494	**0.953**	0.602	**0.6451**	0.7082	**0.7368**	0.9652	**0.9733**
XGB	0.9497	**0.9527**	0.5954	**0.6477**	0.707	**0.7363**	0.965	**0.9738**
RNN-LSTM	0.9491	**0.9529**	0.6131	**0.6327**	0.7106	**0.7328**	0.9648	**0.975**
LSTM	0.9479	**0.9522**	0.5908	**0.6392**	0.6981	**0.7318**	**0.9641**	0.9734

RNN	0.9487	**0.9529**	0.5739	**0.6222**	0.6952	**0.7292**	0.9646	**0.9747**
MLP	0.9449	**0.9498**	0.5288	**0.6078**	0.662	**0.7118**	0.9606	**0.9673**
CNN	**0.9373**	0.9351	**0.4529**	0.3908	**0.5956**	0.5512	**0.9417**	0.9411
Average	0.9470	**0.9504**	0.5698	**0.6046**	0.6856	**0.7092**	0.9616	**0.9691**
Gain	**0.35%**		**6.12%**		**3.45%**		**0.79%**	

4.4. 효율적 모델링 전략 및 SHAP 기반 설명력 검증

메모 포함[K11]: 따라서 제목을 이렇게 변경하고 예측 성능으로 변수 중요도가 나온건 좋은데 교차검증으로 SHAP 결과를 추가하는게 좋을 것 같습니다.

마지막으로, 앞서 분석한 각 변수 조합 환경에서 어떤 알고리즘이 최고의 효율을 발휘하는지, 최적의 알고리즘을 도출하여 비교 분석하였다. <표 4-7>은 4가지 도메인 지식 피처 세트가 적용되었을 때 가장 높은 성능을 보인 모델과 그 지표를 요약한 결과다.

<표 4-7> 도메인 지식 변수 세트별 최적 알고리즘 성능 요약 (Optimal Model Selection)

변수조합	최적알고리즘	Precision	Recall	F1-score	Accuracy	AUC	Radar Chart
행동 최신성·피로도(Recency/Fatigue)	**RF**	**0.9351**	**0.8105**	**0.8683**	**0.9749**	**0.991**	**1.9972**
시점 역동성(Temporal)	**RF**	0.8882	0.7222	0.7967	0.9624	0.9834	1.8078
장기 가치(RFM/BTYD)	**XGB**	0.9105	0.6784	0.7775	0.9604	0.9759	1.7669
콘텐츠 신선도·경로(Novelty/Path)	**RF**	0.8662	0.6516	0.7438	0.9542	0.9743	1.6786

분석 결과, 주목할 만한 사실이 확인되었다. 양질의 피처가 주어졌을 때 가장 높은 예측력을 기록한 모델은 복잡한 연산 구조를 가진 딥러닝이 아니라, 오히려 전통적 머신러닝 기반의 Random Forest(RF)'와 'XGBoost(XGB)'였다.

특히 '행동 최신성 및 피로도(Recency/Fatigue)' 변수가 결합된 환경에서 Random Forest는 F1-score 0.8683, ROC AUC 0.9910을 달성하며 실험된 모든 조합 중 압도적인 1위를 기록하였다. 무엇보다 정밀도(Precision)가 0.9351로 매우 높게 나타났는데, 이는 모델이 "이 고객은 구매할 것이다"라고 예측했을 때 실제 구매로 이어질 확률이 93% 이상임으로 해석된다. 마케팅 비용 낭비의 주범인 오탐(False Positive) 비율을 획기적으로 낮춘 것이다.

반면, 시계열 예측에 특화되었다고 알려진 RNN-LSTM(F1-score 0.8642) 등의 딥러닝 모델은 트리 기반 머신러닝 모델과 대등하거나 소폭 낮은 성과를 보이는 데 그쳤다.

이는 도메인 지식 기반의 피처 엔지니어링(Feature Engineering)이 데이터 사이의 맥락(Context)을 충분히 설명해 준다면, 굳이 연산 비용이 높은 딥러닝을 고집할 필요가 없음을 뒷받침한다. 따라서 학습 시간, 하드웨어 자원(GPU) 소모량, 그리고 운영의 용이성(ROI)을 종합적으로 고려할 때, 실무에서의 최적해는 "무조건적인 딥러닝 도입"이 아닌, "정교한 피처 엔지니어링과 가벼운 머신

러닝의 결합"인 것으로 판단된다.

5. 결론