

애니메이션의 유사도 분석을 통한 온라인 액션게임의 봇 탐지

장창완*, 황준식*, 유태경*

요약

온라인 게임 내의 가상재화인 게임 머니 취득을 목적으로 한 게임 봇의 사용이 게임 서비스에 심각한 문제를 야기하고 있다. 때문에 게임 회사에서도 보안 솔루션을 도입하는 등 여러 가지 대응을 시도하고 있으며 그에 대한 사례로 실제 라이브 서비스에서 게임 봇 탐지에 사용하였던 애니메이션 유사도 분석 프레임워크를 제안한다. 사람과 게임 봇 간에는 애니메이션 패턴 차이가 존재하기에 이를 분석하면 게임 봇을 탐지할 수 있다. 제안하는 프레임워크는 도메인 전문가가 설정한 룰셋에 의존하지 않고 애니메이션 패턴 데이터를 기반으로 생성한 모델을 활용하기에 작업장에서 우회하기가 어려우며, 분석 모델 업데이트만으로 대응이 가능하다는 점에서 게임 운영상 장점이 많다. 이러한 장점으로 향후 애니메이션 패턴 유사도 분석 기법이 게임 봇 탐지에 다양하게 활용될 수 있을 것으로 기대된다.

I. 서론

전 세계 게임 산업은 지속적으로 성장하고 있으며 특히 온라인 게임 내의 가상 사회는 현실 세계와 맞닿아 있는 면이 많아, 여러 분야에서 게임을 이용한 다양한 흥미로운 연구가 진행되고 있다.

현실 세계의 욕구 중 하나인 지배욕과 경쟁의식은 온라인 게임에서도 그대로 드러난다. 특히 성장형 콘텐츠가 주가 되는 RPG 게임에서 플레이어는 경쟁 플레이어 보다 빠르게 성장하여 이를 과시하고자 하는 경향이 있으며, 이를 달성하기 위한 수단으로 게임 내 가상재화인 게임머니에 대한 수요가 높다.

게임머니는 플레이어가 시간과 노력을 투자하여 일정 부분 얻을 수 있으나, 보다 빠르고 편리한 습득을 위해 일부 수요자는 현금으로 이를 취득하려 한다. 이러한 수요는 개인 및 조직적인 공급 세력과 만나 시장을 형성한다. 게임 내 아이템 현금거래 등은 게임이용약관(사용자 계약)과 현행법에서 대부분 금지되어 있음에도 불구하고 게임 머니와 현금 간의 거래 현상이 현실적으로 존재한다.

게임머니 시장에서는 게임 내 재화를 조직적으로 획득하여 현금화시키는 전문적인 ‘작업장 (Gold Farmer Group)’이 등장했다. 과거에는 인건비가 싼 국가의 인

력을 고용하여 게임 머니를 생산하는 형식을 취하다가 최근에는 이를 게임 봇으로 대체해서 자동으로 게임 머니를 생산하는 형태로 고도화 되고 있다.

게임 봇은 사람 대신 게임을 플레이하는 자동화 프로그램이다. 게임 봇은 24시간 동안 플레이하기 때문에 게임 머니와 아이템과 같은 가상 세계의 재화를 일반 사용자보다 더 효율적으로 획득한다.

게임 봇이 아무런 제약 없이 대규모로 활동하게 되면 게임 내 경제 질서와 균형을 파괴하고, 게임 콘텐츠에 대한 일반 플레이어들의 만족감을 떨어뜨린다. 특히 성장에 대한 타인과의 경쟁이 주요 콘텐츠인 MMORPG에서는 공정한 경쟁이 저해되어 플레이어들의 흥미를 떨어뜨리고 이탈하게 만든다. 이러한 이유에서 온라인 게임 회사는 게임 봇으로부터 플레이어를 지키기 위해 여러 가지 노력을 하고 있다.

게임 회사에서 시도하는 대응은 게임 보안 솔루션 배포, 운영 측면에서의 고객 대응, 서버 사이드 로그 분석 등의 노력이 있다.

이 중에서 온라인 액션게임 (예: FPS)의 경우 게임 특성상 Network Latency에 민감한 구조이기에 클라이언트 사이트의 판정에 의존하는 부분이 많다. 때문에 MMORPG에 비해서 상대적으로 게임 보안에 취약한 약점이 있는데 본 연구에서는 서버 사이드 로그 분석

방법으로 애니메이션 패턴의 유사도 분석을 통해서 게임 붓을 분류하는 방법에 대해 설명하였다.

II. 관련 연구

먼저 게임 산업에서 사용하는 게임 붓 탐지 방법을 [표 1]과 같이 분류하였다.

완벽하게 게임 붓을 막을 수 있는 하나의 절대적인 탐지 방법은 존재하지 않으며 각각의 탐지 방법들이 고유의 장점과 단점을 가지고 있다. 현업에서는 한 가지 기법에만 의존하지 않고 여러 방법들을 섞은 복합적인 탐지 레이어를 구성해서 게임 붓에 대응하는 방식을 사용한다.

온라인 게임 런칭 초기에는 보통 전문적인 해킹 대응 솔루션(예:AnhLab 사의 Hackshield 등)에 의존하여 게임 붓의 진입을 막는다. 그러나 라이브 서비스 경력이

오래된 게임일수록 게임 붓이 점차 고도화 되어서 알려진 게임 보안 솔루션만으로는 대응에 한계가 있다. 때문에 게임 콘텐츠의 특징을 활용하여 게임 붓을 사전에 차단하려는 시도가 많이 있어왔다.

그러나 게임 콘텐츠 단에서의 대응은 단기적으로는 효과적으로 작동하나 패턴이 파악되면 우회하기가 쉬운 단점이 있다. 또한 이는 개발사의 지속적인 콘텐츠 대응을 필요로 하므로 개발 부담을 가중시킨다.

최근에는 게임 콘텐츠 단에서의 개발 대응에서 한 단계 나아가서 로그 분석에 기반한 게임 붓 탐지에 대한 접근 방법이 많이 연구되고 있다.

로그 분석을 통한 탐지 기법들의 경우 정확도가 높을 뿐 아니라 원하는 시점에 원하는 만큼 제재할 수 있는 장점이 있다. 아울러 오탐지 이슈나 운영 정책에 따른 로직 수정에도 유연하게 대응할 수 있다. 반면, 데이터

[표 1] 게임 산업에서 사용되는 붓 탐지 방법들

| 구분 | 방법 | 예 | 특징 |
|----------------|--|---|---|
| 보안 전문 솔루션 사용 | 게임 클라이언트의 해킹 및 프로그램 변조 여부를 탐지하는 전문 솔루션 | - NEXON 사의 NGS - 잉카인터넷 사의 GameGaurd - AhnLab 사의 HackShield - Wellbia 사의 XIGNCODE3 | 장점: - 알려진 해킹 프로세스를 탐지하기에 탐지 기준이 직관적이고 고객 대응이 상대적으로 용이함 단점: - 클라이언트 단의 변조 탐지이기에 결국 공격자 측에서 우회 시도가 가능 - 게임 특성에 맞춰서 콘텐츠와 유기적인 결합을 한 탐지가 어려움 |
| 게임 콘텐츠 단에서의 구현 | 게임 클라이언트나 서버에서 특정 조건이나 룰셋 기반 혹은 게임 내 디자인을 이용하여 탐지 및 제재하는 대응 방법 | - NEXON 사의 베이플 스토리에서 활용한 거짓말 탐지기 도입 사례 - 보이지 않는 NPC나 아이템을 이용한 탐지: 사람은 볼 수 없는 아이템을 붓 프로그램은 감지하여 획득함 | 장점: - 프로세스 변조 감지에서 한 단계 더 들어가서 게임 콘텐츠를 이용하여 비정상 행위를 감별 단점: - 게임 서비스 패치 업데이트와 연관이 있기에 개발팀의 콘텐츠 개발 및 라이브 서비스 부담이 증가함 |
| 로그 분석을 활용한 탐지 | 게임 클라이언트와 서버에서 게임 콘텐츠와 관련된 로그를 수집 및 분석해서 탐지하는 방법 | - NEXON 사에서 Splunk로 수집한 게임 로그를 룰셋 기반의 탐지를 한 사례 - NCSOFT 사에서 자기 유사도(Self Similarity)를 사용하여 붓 탐지를 한 사례 | 장점: - 라이브 서비스 업데이트 및 패치 부담이 적고 제재 시점과 로직 업데이트 등 운영 관제에 융통성이 있음 - 로직에 따라서 상호 보완적으로 크로스 체크가 가능하기에 정확도가 높고 오탐지 및 룰셋 노출에 대한 리스크가 적음 단점: - 로깅 작업에 대한 초기 개발 비용이 존재함 - 로그 해상도가 높고 로직이 고도화 될수록 데이터 수집 및 처리 인프라 구성에 따른 리소스 비용 부담이 존재함 |

수집 및 분석 인프라나 기술 확보 등 초기 진입 장벽 및 투자비용이 높은 단점 역시 존재한다. 또한 로그 수집 인프라 개발 비용과 리소스 부담 때문에, 분석에 사용할 로그의 해상도를 높이지 못하는 부차적인 문제도 있다.

그러나 최근 빅데이터 및 데이터마이닝 기술이 발전하고 클라우드 서비스의 보편화로 인프라 도입 비용이 낮아지면서 서버 사이드 로그 분석을 통한 탐지에 대한 연구가 활발히 진행되는 추세이다.

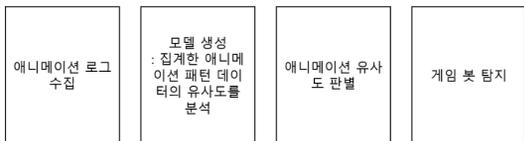
이와 관련된 최근의 연구들을 살펴보면 큰 흐름이 다음과 같은 형태를 띄는 것을 발견할 수 있다. 첫째, 데이터 마이닝과 로그 분석을 사용하여 게임 봇을 탐지하는 연구 비중이 높아지고 있다. 둘째, 기존의 전통적인 접근은 미리 설계한 탐지 룰셋을 바탕으로 봇을 탐지하려는 시도들이 주었던 반면에 최근에는 머신러닝 기법을 활용하여 데이터로부터 직접 결론을 도출하려는 시도가 이루어지고 있다.

본 연구에서는 게임 캐릭터의 애니메이션 시퀀스 데이터를 수집하여 사람과 게임 봇의 패턴간의 유사도를 비교하여 게임 봇을 탐지하는 모델을 제안한다.

Ⅲ. 애니메이션 패턴의 유사도 분석을 활용한 방법론

본 연구에서는 온라인 액션게임에서의 게임 봇 탐지를 위해 정상 유저와 게임 봇 간의 플레이 패턴에 따른 애니메이션 데이터의 유사도를 비교 분석하는 방법을 제안한다.

온라인 액션 게임에서는 플레이어가 게임 캐릭터를 조작하여 이동, 공격, 스킬 사용 등의 애니메이션 액션을 실행한다. 이들이 발생시키는 애니메이션 패턴 데이터를 분석하면 일반적인 플레이를 하는 정상 유저와 자동 반복 사냥 및 클라이언트 번조나 해킹을 시도하는 게임 봇의 행동 패턴에 차이가 존재함을 확인할 수 있다. 이러한 패턴 간의 차이를 유사도로 정량화하여 게임 봇을 탐지할 수 있다.



(그림 1) 애니메이션 유사도 분석 프레임워크 제안

본 연구에서 제안하는 분석 프레임워크는 다음과 같다. 애니메이션 로그 수집 단계에서는 게임 클라이언트/서버에서 애니메이션을 재생한 빈도를 수집한다.

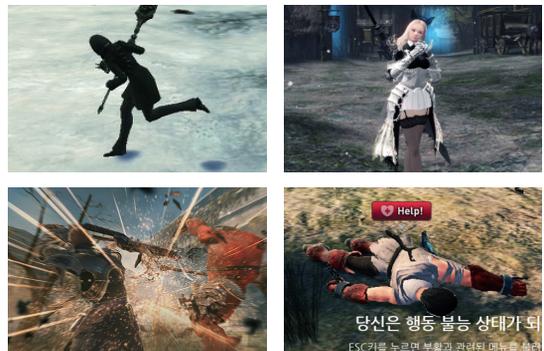
모델 생성 단계에서는 과거의 제재 내역을 기반으로 사람과 봇을 분류해서 해당 애니메이션 패턴의 데이터를 학습한 모델을 생성한다. 실제 새로운 애니메이션 패턴 데이터가 입력되면 해당 모델과의 유사도를 판별해서 최종적으로 게임 봇 여부를 감별한다.

해당 프레임워크를 통한 접근의 유용성을 검증하기 위해 라이브 서비스 8년차의 국내 14위권의 중견 온라인 액션 게임에서 수집된 애니메이션 시퀀스의 재생빈도 데이터를 사용해서 집계하였다.

이렇게 수집한 애니메이션 시퀀스의 재생 빈도를 집계한 결과 일반 플레이어와 게임 봇은 다음과 같은 차이를 보인다.

(표 2) 30초 간격으로 애니메이션 시퀀스 재생 빈도를 집계한 결과 예시

| 구분 | 애니메이션 시퀀스 | 재생 빈도 |
|----|-----------|-------|
| 사람 | 기본 공격 | 20 |
| | 빠르게 달리기 | 100 |
| | 제자리 휴식 | 30 |
| | 행동 불능 | 1 |
| | 피해 입음 | 3 |
| 봇 | 기본 공격 | 140 |
| | 달리기 | 3 |
| | 제자리 휴식 | 300 |
| | 행동 불능 | 0 |
| | 피해 입음 | 3 |



(그림 2) 이동, 제자리 대기, 공격, 행동 불능 등의 캐릭터 조작에 따른 애니메이션 시퀀스 예시

일반 플레이어는 비효율적인 행위(예: 사망, 부활, 사다리 타기 등)를 포함한 다양한 행동을 수행하는 반면에 게임 봇은 재화 습득의 효율성을 극도로 추구하는 양상을 보인다. 게임 봇은 가장 효율이 좋은 특정 행동을 반복 수행하는 경향이 있으며 이동/공격 애니메이션 시퀀스 없이 퀘스트를 클리어 하는 등 맥락상 비정상적인 플레이 패턴을 보인다.

액션 애니메이션을 하나의 Term으로 간주하고 Term의 집합을 Document로 정의한 후 이를 워드 클라우드로 시각화 해 보면 그 결과는 [그림 3]과 같다. 이 때 사람과 게임 봇의 플레이 패턴 사이에는 어느 정도 육안으로 구분되는 특징이 있음을 확인할 수 있었다.

이러한 차이를 명확하게 드러내기 위해서 TF-IDF 가중치를 적용해 보았다.

- $tf(w)$: Term(애니메이션 시퀀스)가 특정 Document (플레이 패턴)에 출현한 빈도
- $df(w)$: Term(애니메이션 시퀀스)가 출현한 전체 Document(플레이 패턴) 수

$$TF-IDF(w) = tf(w) * \log\left(\frac{N}{df(w)}\right) \quad (1)$$

발생 빈도가 낮지만 발생 자체가 유의미한 애니메이션에 변별력을 부여하기 위해 TF-IDF를 사용하였다. 가령, 게임 봇은 ‘이동’과 ‘사다리타기’ 둘 다 안하는 반면에 일반 플레이어는 ‘이동’은 하지만 ‘사다리타기’는 가끔 실행한다.

실행 빈도의 관점에서 보면 ‘이동’은 봇과 사람을 구분할 수 있는 중요한 피처인 반면에 ‘사다리타기’는 상대적 중요도가 떨어진다. 그러나 ‘사다리타기’는 게임 맥락 및 재화 습득 효율성의 관점에서 ‘이동’보다 더 비



[그림 3] 사람과 봇의 애니메이션 패턴을 워드 클라우드로 시각화

[표 3] 애니메이션 시퀀스 빈도값에 TF-IDF를 적용한 결과 값 TOP 5

| 구분 | 애니메이션 시퀀스 | 재생 빈도 |
|----|--------------|-------|
| 사람 | 특수 스킬 사용 | 3.78 |
| | 감정표현 | 3.50 |
| | 주변 환경과 상호 작용 | 2.80 |
| | 발사체 사용 | 2.49 |
| | 보스에게 붙잡힘 | 1.31 |
| 봇 | 후방에서 가벼운 피해 | 1.50 |
| | 정면에서 강한 피해 | 1.49 |
| | 피해 경직 | 1.09 |
| | 제자리 휴식 | 0.80 |
| | 제자리 대기 | 0.76 |

효율적인 ‘인간적’ 행위일 수 있다. 이처럼 발생 빈도가 낮지만 중요도가 높은 항목에 가중치를 부여하기 위해서 TF-IDF를 사용하였다. 그 결과는 [표 3]과 같다.

TF-IDF 가중치 적용한 결과를 워드 클라우드로 시각화한 결과는 [그림 4]와 같다. 일반 플레이어와 게임 봇의 플레이 패턴에 따른 애니메이션 시퀀스의 차이가 더욱 뚜렷해지는 것을 알 수 있다.

이를 Bar Chart로 시각화 해 보면 [그림 5]와 같다. [그림 5]에서 볼 수 있듯이 사람들의 평균과 게임 봇 사이에는 확연한 차이가 있다.

이러한 애니메이션 패턴 특성의 차이점을 활용하여 사람과 봇을 분류하는 방법은 다음과 같다.

과거 제재기록을 활용하여 일반 플레이어와 게임 봇의 애니메이션 데이터를 기록한 정답지를 사전에 준비한다. 정답지 중에서 새로운 애니메이션 데이터와 가장 유사한 10개를 추출한다. 추출한 정답지 10개 중에서 9개 이상이 게임 봇이라면 새로운 유저를 봇으로 판정한다.



[그림 4] 사람과 봇의 애니메이션 패턴의 TF-IDF 값을 워드 클라우드로 시각화



(그림 5) 애니메이션 패턴 데이터의 TF-IDF 값을 Bar Chart 로 표시해서 비교

이 때 애니메이션 패턴 데이터의 유사도를 판단하는 기준은 Cosine Similarity를 사용한다. Cosine Similarity의 계산은 (2)와 같다.

$$\begin{aligned}
 \text{Similarity} &= \cos(\theta) \\
 &= \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}} \quad (2)
 \end{aligned}$$

해당 분류 방법을 적용하여 실제로 게임 봇과 애니메이션 패턴이 매우 유사한 캐릭터들을 찾아낼 수 있었으며 새로 찾아낸 유사한 애니메이션 패턴의 캐릭터들은 수동 검수결과 모두 게임 봇으로 확인되었다.

```

import pandas as pd
from sklearn.feature_extraction.text import TfidfTransformer
from scipy.spatial.distance import cosine

# 초기화(term_doc_matrix & tfidf 생성)
def init():
    docs, labels = load_trn_anim()
    term_doc_mat = create_term_doc_matrix(docs)
    tfidf = create_tfidf(term_doc_mat)
    return (docs, labels, term_doc_mat, tfidf)

# term & doc 빈도 값을 담은 matrix 생성
def create_term_doc_matrix(docs):
    term_doc_matrix = pd.DataFrame(docs, T)
    term_doc_matrix.fillna(0, inplace=True)
    return term_doc_matrix

# term & doc 빈도의 tf-idf 가중치 계산
def create_tfidf(term_doc_matrix):
    tfidf = TfidfTransformer(norm='l2', use_idf=True, smooth_idf=True, sublinear_tf=True).fit_transform(term_doc_matrix.values)
    tfidf = pd.DataFrame(tfidf.todense())
    return tfidf

# 유사도 분석
def test_similarity(tst_docs, trn_tfidf, term_doc_mat, labels):
    tst_df = pd.DataFrame(tst_docs, T)
    tst_df = tst_df.reindex(columns=term_doc_mat.columns, fill_value=0)
    tst_df.fillna(0, inplace=True)
    tst_tfidf = TfidfTransformer(norm='l2', use_idf=True, smooth_idf=True, sublinear_tf=True).fit_transform(tst_df.values)
    tst_tfidf = tst_tfidf.todense()

# cosine 유사도 랭킹추출
cids = []
scores = []
neighbors = []

for i in range(tst_tfidf.shape[0]):
    check_similar = lambda x: 1 - cosine(tst_tfidf[i], x)
    similars = trn_tfidf.apply(check_similar, axis=1)
    results = pd.DataFrame({'CID':term_doc_mat.index, 'Similar':similars, 'Label':labels})

# 스코어 집계
results = results.sort_values(['Similar'], ascending=False)[:10]
filter = (results['Label'] == 'abuse')
cnt = results[filter]['Label'].count()
mean = results[filter]['Similar'].mean()
score = cnt * mean if cnt > 0 else 0

cids.append(tst_df.index[i])
scores.append(score)
neighbors.append(results)

return cids, scores, neighbors

```

(그림 6) 애니메이션 유사도 분석 Python 스크립트

IV. 실 험

위의 프레임워크를 앞서 언급하였던 애니메이션 시퀀스 데이터를 수집하였던 온라인 액션 게임에 적용해 보았다. 이 때 라이브 서비스 중인 게임 서비스의 특성 상 오탐지와 운영 부담을 최소화하기 위하여 표본 크기를 점진적으로 늘려가면서 검수를 진행하였다.

특정 시점에 특정 퀘스트를 완료한 캐릭터를 대상으로 애니메이션 유사도 분석 기법을 적용했을 때 1차로 300건, 2차로 150건의 게임 봇을 탐지하였으며, 해당 캐릭터들은 100% 봇인 것으로 판명되었다.

이후 동일한 프레임워크를 사용하여 게임 봇으로 의심되는 캐릭터들을 탐색해 보았다. 그 결과 특정 국가 서비스의 일간 전투 완료 캐릭터 중 약 9% 정도가 아직 알려지지 않은 게임 봇으로 추정되었으며 특히 중국 서비스의 경우는 게임 봇의 비중이 다른 국가보다 더 높은 것으로 추정되었다.

V. 결 론

본 연구에서는 온라인 액션게임에서 애니메이션 패턴의 유사도 분석을 활용하여 게임 봇을 탐지하는 프레임워크를 사용하여 게임 봇을 탐지하였다.

본 연구에서 제안하는 탐지 방법은 도메인 전문가가 설정한 룰셋에 의존하지 않고 애니메이션 패턴 데이터를 기반으로 생성한 모델을 활용하기 때문에, 작업장에서 탐지 패턴을 분석하여 우회하기가 어려운 장점이 있다. 또한 게임 봇 탐지 성능이 떨어지는 경우에도 모바일 게임을 패치하지 않고 분석 모델 업데이트만으로 대응이 가능하다는 점에서 게임 운영과 조화를 이룬다. 이러한 장점들로 인해 향후 게임 봇 탐지에 있어서 애니메이션 패턴 유사도 분석 기법이 적극적으로 활용될 수 있을 것으로 기대된다.

참 고 문 헌

- [1] 이은석, “pay-to-skip: 온라인 게임속 로봇 경제와 내몰리는 인간”, NDC 2016
- [2] A.R.Kang, J.Woo, J.Park, and H.K Kim, “User Behavior Analysis for Online Game Bot Detection.” Journal of The Korea Institute of Information Security & Cryptology. vol. 22, no 2. pp.225-328, Apr. 2012
- [3] Eunjo Lee, Jiyoung Woo, Hyoungshick Kim, Aziz Mohaisen, Huy Kang Kim. “You are a Game Bot: Uncovering Game Bots in MMORPGs via Self-similarity in the Wild NDSS2016”
- [4] 광병일, 김휘강, “온라인 게임에서의 이상 징후 탐지 기법 조사 및 분류”, Journal of The Korea Institute of Information Security & Cryptology vol.25, NO.5, Oct.2015
- [5] 김하나, 광병일, 김휘강, “MMORPG 게임 내 계정 도용탐지 모델에 관한 연구”, Journal of The Korea Institute of Information Security & Cryptology vol.25 NO.3, Jun.2015
- [6] 김선민, 김휘강. “FPS 게임 서버 로그 분석을 통한 클라이언트 단 치팅 탐지 기능 개선에 관한 연구”, Journal of The Korea Institute of Information Security & Cryptology vol.25, NO.6, Dec.2015
- [7] 강성욱, 이진, 이재혁, 김휘강. “MMORPG에서 GFG 쇠퇴를 위한 현금거래 구매자 탐지 방안에 관한 연구”, Journal of The Korea Institute of Information Security & Cryptology vol.25, NO.4, Aug.2015
- [8] 정성훈, 김하나, 신영상, 이태진, 김휘강. “결제로 그 분석 및 데이터 마이닝을 이용한 이상거래 탐지 연구조사”, Journal of The Korea Institute of Information Security & Cryptology vol.25, NO.6, Dec.2015
- [9] 김하량, 김휘강, “온라인 게임 봇 길드 탐지 방안 연구”, Journal of The Korea Institute of Information Security & Cryptology, vol.25, NO.5, Oct.2015
- [10] 송현민, 김휘강, “자산변동 좌표 클러스터링 기반 게임봇 탐지”, Journal of The Korea Institute of Information Security & Cryptology
- [11] 이진, 강성욱, 김휘강, “캐릭터 성장 유형 분류를 통한 온라인 게임 하드코어 유저와 게임 봇 탐지 연구”, Journal of The Korea Institute of Information Security & Cryptology, vol.25, NO.5, Oct.2015

〈저자 소개〉



장창완 (Changwan Jang)

2007년 2월 : 명지대학교 컴퓨터공학과 졸업

2009년 9월~2016년 4월 : 넥슨코리아 <마비노기 영웅전> 개발실

2016년 5월~현재 : 넥슨코리아 분석본부 개발팀

관심분야 : 빅 데이터 분석, 데이터 엔지니어링



유태경 (Taekyung Yoo)

2013년 2월 : 부산대학교 정보컴퓨터공학부 졸업

2015년 2월 : UNIST 컴퓨터공학부 졸업

2016년 11월~현재 : 넥슨코리아
관심분야 : 대용량 데이터 분산 처리, 데이터 엔지니어링



황준식 (Junsik Hwang)

2016년 7월~현재 : 넥슨코리아 분석본부 데이터분석팀

관심분야 : 데이터 분석, 머신 러닝