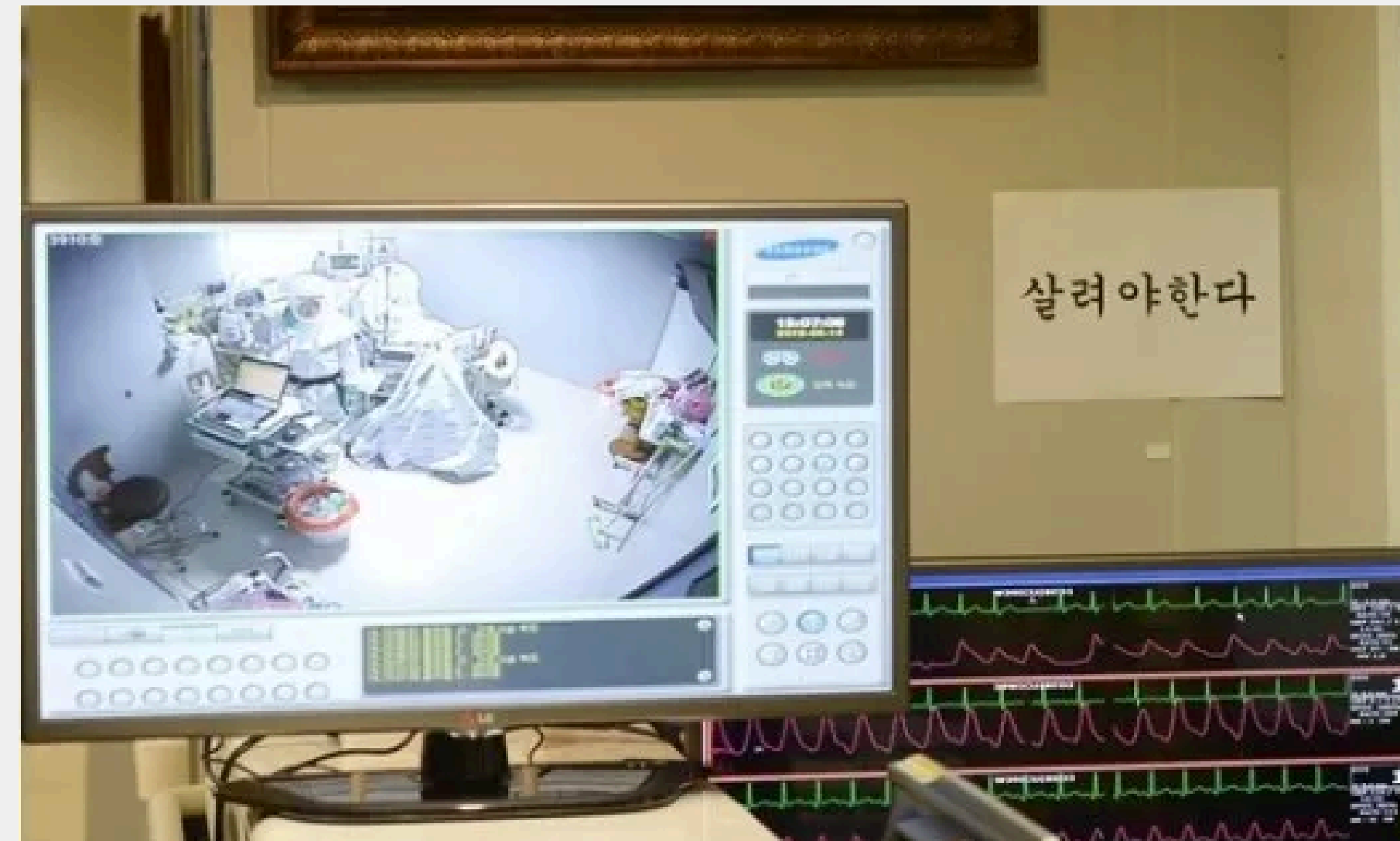


# 살려야한다

질병 프로세스를 이용한 최적의 의료 처치 제안

6조 | 박찬혁 부형진 정수진 정필규



# Content

01 주제 선정 이유

---

02 테이블 설명

---

03 사용 방안

---

04 프로세스 설명

---

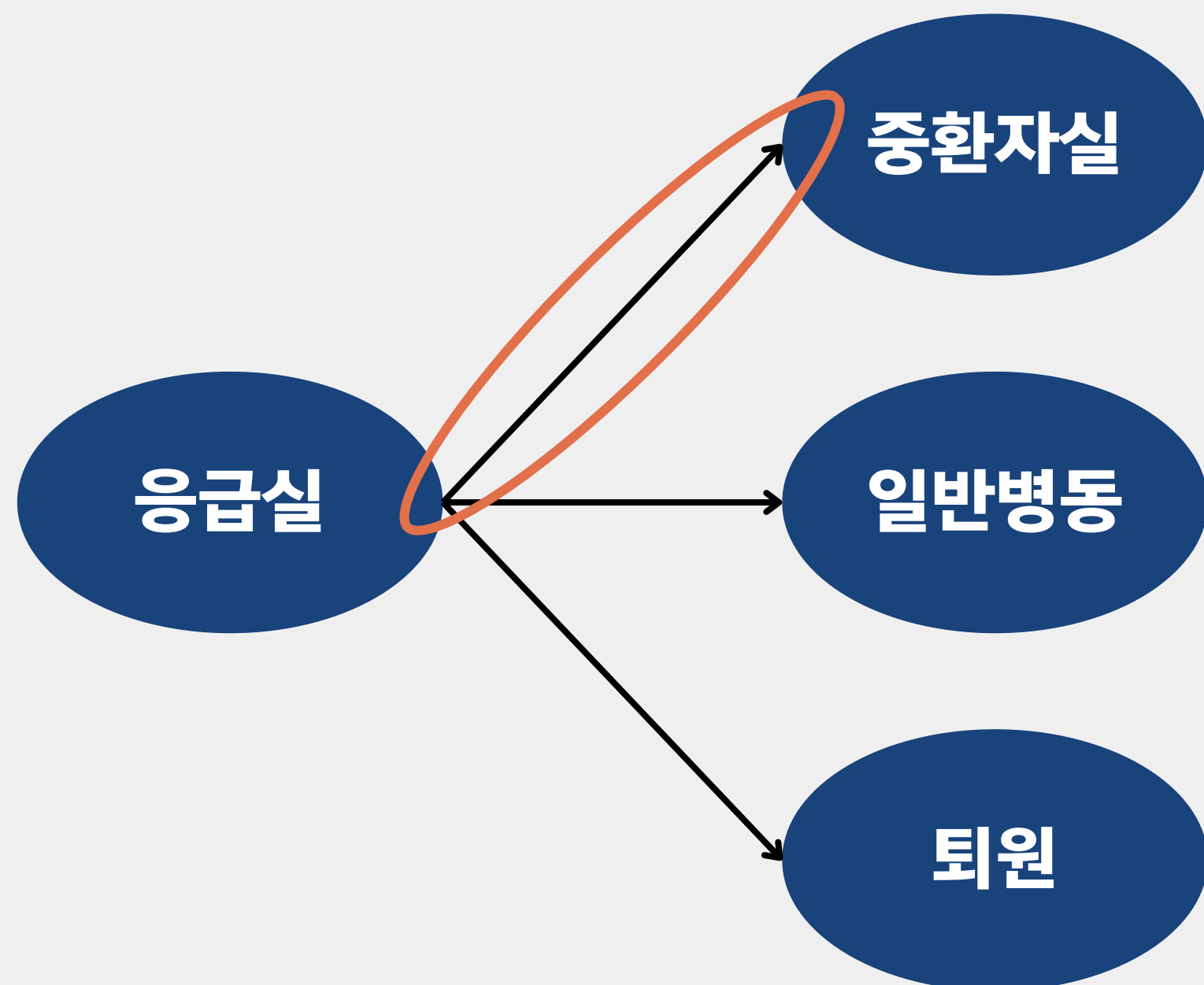
# 01

## 주제 소개

사회 의료·건강

**'1시간내 수술 필요' 중환자, 응급실  
체류만 4시간 넘어**

**"중환자실은 응급실 연장선...응급의학과 영역 확대  
도전"**



응급실 입원 환자 중 가장 빠른 처치가 필요한 환자는 중환자실  
입원 예정자

응급실과 중환자실은 통합적으로 유지될 필요 있음

응급실은 응급의학과 소관, 중환자실은 개별 의학과 담당으로  
환자의 통합적 관리에 어려움 있음

응급실과 중환자실의 환자 이동 경로를 통합적으로 분석하는  
프로세스를 모델링하고자 함

# 01

## 주제 소개

### 병원 운영 측면

---

병상 스케줄 관리에 기여

응급실과 중환자실의 통합적 관리 가능

### 환자 측면

---

개별 환자에게 최적화된 의료 프로세스를 제공

중환자실과 응급실의 trace를 최적화하여 환자의 생존 확률을 높일 수 있음

# 02 테이블 설명

## HOS

---

### **d\_icd\_diagnoses/ d\_icd\_procedures**

의학적 진단명과 의료 절차를 국제 표준 ICD 코드로 나타낸 테이블

## ICU

---

**d\_items** : item\_id에 따른 추가 정보

**Procedureevents** : 중환자실에서 수행된 의학적 절차와 처치 정보

**chartevents** : 환자 모니터링 시스템이 기록한 다양한 생체 신호

## ED-log-data

---

응급실 내에서 환자의 상태 변화와 관련된 중요한 이벤트를 기록하며, 응급실에서의 환자 경로와 치료 과정에 대한 세부적인 정보를 제공하는 이벤트 로그

## CORE

---

### **admissions**

환자의 병원 방문 시점부터 퇴원 시점까지의 정보

# 03 사용 방안

## HOS

---

### **d\_icd\_diagnoses/ d\_icd\_procedures**

icd 진단 코드를 큰 범주로 나누어 (ex) 280-289 혈액 및 조혈 기관의 질환) 질병 구분에 사용

## ICU

---

**d\_items** : linksto로 관련 있는 테이블 확인 후 item\_id를 통해 병합

**procedureevents** : 중환자실의 의료 절차, 진단 기기, 수술 및 의료 기구 등 정보 확인

**chartevents** : warning 표시를 이용해 경고 확인

## ED-log-data

---

응급실에서 부터 중환자실 까지 이어지는 트레이스 분석에 사용

## CORE

---

### **admissions**

admittime/dischtime 환자의 입원/퇴원 시간 확인

# 04 프로세스 설명

가장 빈번하게 발생하는  
질병 카테고리 탐색

각 진단들을 카테고리로 묶은 후  
응급실에서 중환자실로 이동하는  
환자들의 진단명을 종합하여  
빈번하게 발생하는 카테고리를  
수집

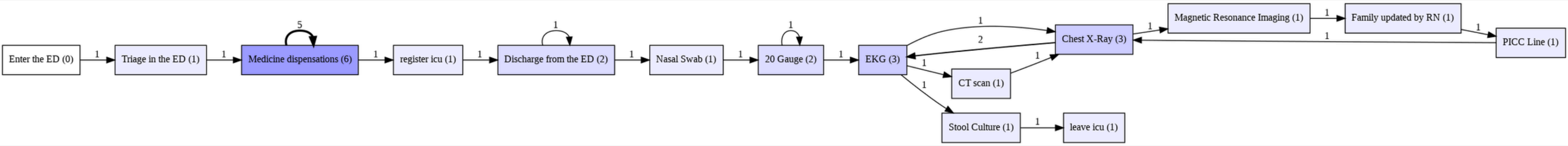
질병 치료 프로세스  
분석 후 모델 생성

중환자실에서 이뤄지는 처치를 사망자  
와 완치자를 비교해 딥러닝 모델을 통해  
분석하고, 어떤 trace가 가장 최적인지  
예측하는 모델 생성

실제 프로세스와의  
비교, 최적의 방안  
제안

실제 응급실부터 중환자실까지  
이어지는 프로세스 중 가장 빈번하게  
발생하는 프로세스와 최적으로  
예측된 값을 비교하여 어떤 구간이  
문제점인지를 제안

# 응급실 입원부터 중환자실 입원 후 퇴원까지의 trace 시각화





# 05 사용 예정 모델



## 딥러닝 모델

- LSTM  
각 의료 처치가 시계열 데이터의 형태를 띄고 있기 때문에 각 생체 신호의 변화에 따라 예측 모델을 구성
- Attention Mechanism  
특정 시점에서 질병이 완화되는 경우가 존재할 수 있기 때문에 이를 예측에 더 중요한 정보로 인식하게 하기 위한 방안

## 프로세스 모델

- Petri Net
- Process tree

두 방안 모두 각 루트의 빈도수를 확인하기 좋음



# Comming Soon

감사합니다