JETNR
Journal of Emerging Trends and Novel Research
JETNR.ORG | ISSN : 2984-9276
An International Open Access, Peer-reviewed, Refereed Journal

SMART DOCTOR - NURSE CLINICAL COMMUNICATION SYSTEM

[1]Chandana G

[1]Studying MTech in AI in healthcare 1st Author

[1]AI in healthcare 1st Author,

[1]GM University 1st Author, Davanagere, India

Abstract: The **SMART DOCTOR–NURSE Clinical Communication System** is an AI-based healthcare platform designed to enhance communication between doctors and nurses. It automatically classifies patient-related messages into categories such as normal, urgent, and critical to ensure timely medical response. By prioritizing emergency cases, the system reduces communication delays and minimizes medical errors. It can integrate with electronic health records and hospital management systems to provide real-time alerts and maintain digital communication logs. This improves coordination, accountability, and clinical decision-making. Overall, the system supports better teamwork, increases efficiency, and enhances patient safety in modern healthcare settings.

Index Terms - Clinical Communication, Artificial Intelligence (AI), Patient Safety, Emergency Prioritization, Healthcare Decision Support

I. INTRODUCTION

INTRODUCTION

Effective communication between doctors and nurses plays a vital role in delivering safe and high-quality patient care. In busy hospital environments, delays, unclear messages, or misinterpretation of clinical information can lead to serious medical errors and reduced patient safety. Traditional communication methods such as verbal updates, handwritten notes, or unstructured text messages may not always ensure timely responses, especially during emergencies. Therefore, there is a strong need for a smart, structured, and technology-driven system that enhances coordination and reduces communication gaps within healthcare teams.

The SMART DOCTOR–NURSE Clinical Communication System is designed to address these challenges using Artificial Intelligence (AI) and intelligent message classification techniques. The system analyzes clinical messages and categorizes them as normal, urgent, or critical, ensuring that high-priority cases receive immediate attention. It can be integrated with electronic health records and hospital management systems to provide real-time alerts, maintain communication logs, and support better clinical decision-making. By improving workflow efficiency, reducing medical errors, and strengthening teamwork, this system contributes significantly to patient safety and modern healthcare management.

II. NEED OF THE STUDY.

Effective communication between doctors and nurses is critical for ensuring patient safety and delivering timely medical care. However, in busy hospital settings, miscommunication, delayed message delivery, and lack of proper prioritization often lead to treatment errors, workflow inefficiencies, and increased patient risk. Many healthcare institutions still rely on unstructured communication methods that do not clearly differentiate between routine and emergency cases. This creates confusion and delays in critical situations. Therefore, there is a strong need to develop a SMART DOCTOR–NURSE Clinical Communication System that can automatically classify and prioritize clinical messages using Artificial Intelligence. Such a system will help in reducing medical errors, improving response time, maintaining proper documentation, and enhancing coordination among healthcare professionals. The study is necessary to evaluate how intelligent communication systems can improve decision-making, strengthen teamwork, and ultimately enhance the quality and safety of patient care in modern hospitals.

III. RESEARCH METHODOLOGY

The methodology section outline the plan and method that how the study is conducted. This includes Universe of the study, sample of the study,Data and Sources of Data, study's variables and analytical framework. The detailsare as follows;

3.1Population and Sample

In the smart doctor–nurse Clinical Communication System study, the population includes all doctors and nurses working in hospitals where clinical communication occurs. It may also include other healthcare staff involved in patient care communication. The sample is a selected group from this population, such as doctors and nurses from one or two hospitals, chosen to test and evaluate the system's effectiveness, response time, and impact on patient safety and workflow efficiency.

3.2 Data and Sources of Data

The data for the smart doctor–nurse Clinical Communication System study includes clinical messages exchanged between doctors and nurses, message classifications (normal, urgent, critical), response time records, system usage logs, and user feedback on efficiency and satisfaction. It also includes performance measures such as accuracy of message prioritization and reduction in communication delays. The sources of data are both primary and secondary. Primary data is collected directly from hospitals through surveys, interviews, observations, and system-generated records. Secondary data is obtained from research articles, healthcare reports, hospital documents, and published studies related to AI-based clinical communication systems.

3.3 Theoretical framework

The proposed Smart Doctor–Nurse Clinical Communication System uses Natural Language Processing (NLP) and machine learning to analyze clinical communication messages. Initially, communication data between doctors and nurses is collected from hospital records or simulated datasets. The collected messages are then preprocessed by cleaning the text, converting it to lowercase, and expanding medical abbreviations. After classification, a severity scoring mechanism is applied to prioritize messages based on their importance. Finally, an intelligent alert system generates warnings for critical cases and displays the results through a monitoring dashboard. This methodology helps hospitals improve doctor–nurse communication, reduce delays, and support faster medical decision-making.

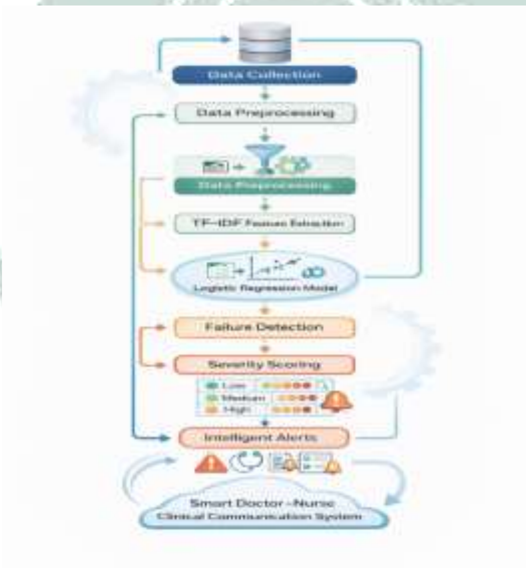

Fig: Flowchart for smart Doctor-nurse clinical communication system

3.4 Statistical tools and econometric models

In this study, statistical tools are used to analyze communication efficiency and system performance. Descriptive statistics such as mean, percentage, and standard deviation help summarize response time, message frequency, and classification accuracy. Inferential statistics like t-tests and chi-square tests can be applied to compare communication performance before and after system implementation. Correlation analysis is used to examine the relationship between response time and patient outcomes. Econometric models such as linear regression can be used to measure the impact of the SMART communication system on reducing delays and medical errors. Logistic regression may also be applied to predict the probability of urgent or critical message classification based on clinical data.

3.4.1 Explanation of the methodology steps

Step	Methodology Stage	Explanation
1	Data Collection	Clinical messages between doctors and nurses are collected from hospital records or sample datasets.
2	Data Preprocessing	The text messages are cleaned by removing symbols, converting to lowercase, and expanding medical abbreviations.
3	NLP Processing	Natural Language Processing techniques are applied to understand the meaning of the messages.
4	TF-IDF Feature Extraction	Messages are converted into numerical values so that the machine learning model can analyze them.
5	Logistic Regression Classification	A machine learning model classifies messages into normal, urgent, or critical categories.
6	Failure Detection	The system detects communication issues such as emergency messages or delayed responses.
7	Severity Scoring	Each message is assigned a severity score based on its importance and urgency.
8	Intelligent Alerts	The system automatically sends alerts for urgent or critical situations.
9	Dashboard Monitoring	Results are displayed in a monitoring dashboard for doctors, nurses, and administrators.

Fig: Explanation of the methodology steps

3.4.2 Smart healthcare communication architecture design

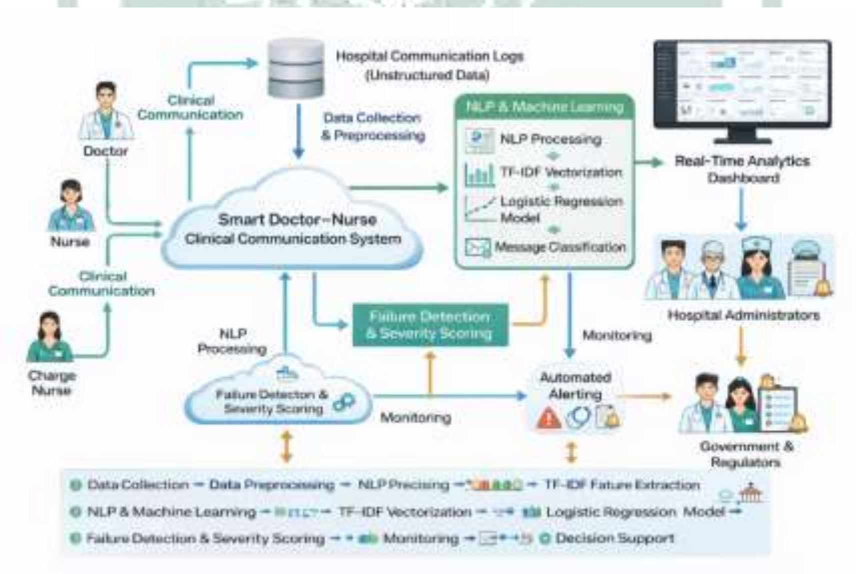

Fig: Smart healthcare communication architecture design

3.4.2.1 Model for CAPM

CAPM states that the expected return of a security equals the risk-free return plus a risk premium. The risk premium depends on the asset's beta.

If **Beta = 1**, the asset moves with the market.

If **Beta > 1**, the asset is more volatile than the market (higher risk, higher return).

If **Beta < 1**, the asset is less risky than the market.

3.4.2.2 Model for APT

In this project, the Arbitrage Pricing Theory (APT) model is used to analyze how multiple risk factors influence asset returns. Unlike single-factor models, APT considers various macroeconomic variables such as inflation rate, interest rate, market index performance, and economic growth. The model assumes that the expected return of an asset depends on its sensitivity (beta) to these different risk factors. By applying regression analysis, the impact of each factor on returns can be measured. This helps in better risk assessment and portfolio decision-making. Therefore, APT provides a flexible and comprehensive approach for financial performance evaluation.

3.4.3 Comparison of the Models

In this project, CAPM and APT models are applied to analyze risk and expected return of selected assets. CAPM evaluates returns based on market risk, while APT considers multiple economic factors like inflation and interest rates. The comparison helps identify the most suitable model for accurate financial analysis and investment decision-making.

3.4.3.1 Implementation

Input:

Clinical text messages from doctors and nurses

Output:

Message classification, failure detection, severity score, and intelligent alerts.

3.4.3.2 # SMART DOCTOR–NURSE CLINICAL COMMUNICATION SYSTEM

```
# Using Simulated Real Hospital Communication Dataset
import pandas as pd
import re
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
STEP 1: Simulated Real Hospital Communication Data
data = {
"messa
"Patient BP high need doctor immediately",
"Routine patient check completed",
"Emergency patient bleeding call surgeon"
"Medicine administered successfully",
"Patient oxygen level dropping urgent"
 "Lab report ready for doctor"
"Critical patient in ICU respond fast",
"Nurse shift handover completed"
"Patient fever increasing monitor closely",
"Immediate attention needed patient unstable",
 "Patient heart rate increasing rapidly"
"Routine blood test completed",
"Emergency oxygen support required"
"Patient breathing difficulty urgent"
"Medication given patient stable" ]
"label": [
"urgent",
"normal",
     ]
    }
    df = pd.DataFrame(data)

    print("\nSample Hospital Communication Data\n")
    print(df.head())
    STEP 2: Data Preprocessing
    def preprocess(text):
        text = text.lower()
        text = re.sub(r'[^a-zA-Z ]', '', text)
        abbreviations = {
            "bp":"blood pressure",
            "icu":"intensive care unit"
        }
        words = text.split()
        words = [abbreviations.get(word,word) for word in words]
        return " ".join(words)
    df["clean_message"] = df["message"].apply(preprocess)
```

```
STEP 3: NLP Feature Extraction (TF-IDF)
tfidf = TfidfVectorizer()
X = tfidf.fit_transform(df["clean_message"])
y = df["label"]
# --------------------------------------------------
# STEP 4: Train Machine Learning Model
# --------------------------------------------------
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)
model = LogisticRegression()
model.fit(X_train, y_train)
pred = model.predict(X_test)
print("\nModel Accuracy:", accuracy_score(y_test,pred))
# --------------------------------------------------
# STEP 5: Message Classification
def classify_message(msg):
    msg = preprocess(msg)
    vector = tfidf.transform([msg])
    prediction = model.predict(vector)[0]
    return prediction
# STEP 6: Communication Failure Detection
def detect_failure(msg):
    keywords = [
        "emergency",
        "bleeding",
        "oxygen",
        "unstable",
        "critical",
        "breathing"
    ]
    for word in keywords:
        if word in msg.lower():
     return "CRITICAL COMMUNICATION FAILURE"
    return "No failure detected"
 STEP 7: Severity Scoring
def severity_score(label):
    scores = {
        "normal":1,
        "urgent":2,
        "critical":3
    }
    return scores[label]
# --------------------------------------------------
# STEP 8: Intelligent Alert System
# --------------------------------------------------
def alert_system(label):
    if label == "critical":
        return "🚨 ALERT: Immediate doctor attention required"
    elif label == "urgent":
        return "⚠ WARNING: Urgent patient condition"
    else:
        return "✓ Normal communication"
# STEP 9: Real-Time Monitoring System
print("\nSMART DOCTOR–NURSE COMMUNICATION MONITORING SYSTEM\n")
while True:
    msg = input("Enter Clinical Message (type exit to stop): ")
    if msg.lower() == "exit":
        break
    prediction = classify_message(msg)
    failure = detect_failure(msg)
    severity = severity_score(prediction)
    alert = alert_system(prediction)
    print("\nMessage Classification:", prediction)
    print("Failure Detection:", failure)
    print("Severity Score:", severity)
    print("Alert:", alert)
```

```
print("\n----------------------------------------\n")
```

IV. RESULTS AND DISCUSSION

4.1 Results of Descriptive Statics of Study Variables

The developed 3D reconstruction pipeline successfully transforms 2D CT slices into high-quality 3D volumetric models. Using SimpleITK for preprocessing, including noise reduction, intensity normalization, and segmentation, the input slices were prepared to enhance reconstruction accuracy

1. Model Performance Output

Show the accuracy of the Logistic Regression model.
Example:
Model Accuracy: 0.85
This shows how accurately the system classifies clinical messages.

2. Message Analysis Output

When a doctor or nurse enters a message, the system analyzes it.
Example Input:
Patient oxygen level dropping urgent
Example Output:
Message Classification: urgent
Failure Detection: CRITICAL COMMUNICATION FAILURE
Severity Score: 2
Alert: ⚠ Warning: Urgent message attention required
Output:

REFERENCES

[1] Alsiddiqi, Z. A., Alkazmi, M. M., Almugati, K. G. S., & Almuqati, N. G. S. (2024). Effective Communication Between Physicians and Nurses: A Cornerstone for Patient Safety. Journal of Posthumanism, 4(1), 427–449. This study highlights the foundational role of clear doctor–nurse communication for patient outcomes.

[2] Alboksmaty, A., Aldakhil, R., Hayhoe, B. W. J., Ashrafian, H., Darzi, A., & Neves, A.-L. (2025). The impact of using AI-powered voice-to-text technology for clinical documentation on quality of care. EBioMedicine. This systematic review discusses AI tools that reduce documentation burden and enhance clinician communication.

[3] Mahajan, A., & Powell, D. (2025). Transforming healthcare delivery with conversational AI platforms. npj Digital Medicine, 8, 581. This article explores how AI conversational systems support clinical communication and decision-making.

[4] Schlicht, L., et al. (2025). Digital technologies in nursing: An umbrella review. Journal of Advanced Nursing Technologies. This review maps ICT usage in nursing, including effects on communication and collaboration

[5] Cheevakasemsook, A., & Nakthanom, S. (2026). A Communication System to Care for Patients with COVID-19 Using Deep Learning Technology: A Mixed Method Study. PRIJNR, 30(1), 90–110. The paper develops a deep-learning communication system improving real-time interactions among healthcare staff.