Netflix Movie Recommender System

Using Natural Language Processing and Cosine Similarity

Name: Gaurav Kumar
Student Id: INTERNSHIP_171273103266163398a8c87
Group no.: B1 G4
edunet Mail: gauravk8271@edunetmail.com
personal mail : gauravk8271@gmail.com
Institute name: IIT kharagpur
phone: 6204780646

edunet
foundation

OUTLINE

- Introduction
- Dataset Overview
- Data Preprocessing
- Feature Extraction
- Model Training
- Recommendations
- Conclusion
- Future Scope
- References

PROBLEM STATEMENT

Objective: Develop a movie recommender system to suggest movies based on user preferences.

Importance: Enhances user experience by providing personalized movie recommendations.

Applications: Used by streaming services like Netflix, Amazon Prime, and Hulu to improve user engagement and satisfaction.

PROPOSED SOLUTION

Develop a content-based movie recommender system using natural language processing (NLP) and cosine similarity to suggest movies based on descriptions and genres.

Key Steps:

1. **Data Collection:**
   - Use a dataset with `id`, `title`, `overview`, and `genre`.
2. **Data Preprocessing:**
   - Handle missing values and ensure consistency.
   - Create `tags` by combining `overview` and `genre`.
3. **Feature Extraction:**
   - Convert `tags` into numerical vectors using CountVectorizer.
4. **Model Training:**
   - Compute cosine similarity between movie vectors.
5. **Recommendation System:**
   - Implement a function to recommend movies based on similarity scores.
6. **Evaluation and Improvement:**
   - Assess recommendation accuracy.
   - Explore enhancements like additional features and advanced NLP techniques.
7. **Deployment:**
   - Save the model and similarity matrix using `pickle`.
   - Develop a user interface for real-time recommendations.

Benefits:

- Personalized movie recommendations.
- Enhanced user experience.
- Scalable for large datasets and real-time use.

Dataset overview

- **Source:** Movie metadata dataset containing information about various movies.
- **Key Columns:** `id`, `title`, `overview`, `genre`.
- **Initial Exploration:**
    - Displayed the first 10 records to understand the structure of the data.
    - Descriptive statistics to get an overview of the dataset.
    - Checked for missing values and data types to ensure data quality.

Data Processing

Column Selection: Focused on `id`, `title`, `overview`, and `genre`.

Feature Engineering:

- Created a new feature `tags` by combining `overview` and `genre`.
- This new feature helps in capturing the essence of the movie plot and genre.

Column Dropping: Removed the original `overview` and `genre` columns to simplify the dataset.

Feature Extraction

Vectorization Technique: Used CountVectorizer to convert text data into numerical vectors.

- **Parameters:** `max_features=10000` (limits the number of features to 10,000 most frequent words), `stop_words='english'` (removes common English stop words).

Transformation: Converted the `tags` feature into vectors.

Result: Obtained a high-dimensional sparse matrix representing the textual data.

Model Training

Similarity Measure: Used cosine similarity to measure the similarity between movies.

Cosine Similarity: Measures the cosine of the angle between two vectors, capturing the similarity in their direction.

Implementation: Calculated the cosine similarity between all movie vectors to find similar movies.

Recommendations

Example Process:

- Identified the index for a specific movie (e.g., "The Godfather").
- Sorted movies based on similarity scores to find the most similar movies.

Function Implementation:

- Developed a `recommand` function to recommend movies based on a given title.
- Example Output: Provided recommendations for "Iron Man".

Demonstration: Displayed the top 5 similar movies for the given example

RESULT

Key Results:

1. **Data Preprocessing:** Combined `overview` and `genre` into `tags`.
2. **Feature Extraction:** Converted `tags` to vectors using CountVectorizer.
3. **Model Training:** Calculated cosine similarity between movie vectors.
4. **Recommendation System:** Recommended movies based on similarity scores.
   - Example: For "Iron Man," recommendations included "Iron Man 2," "The Avengers," and "Iron Man 3."

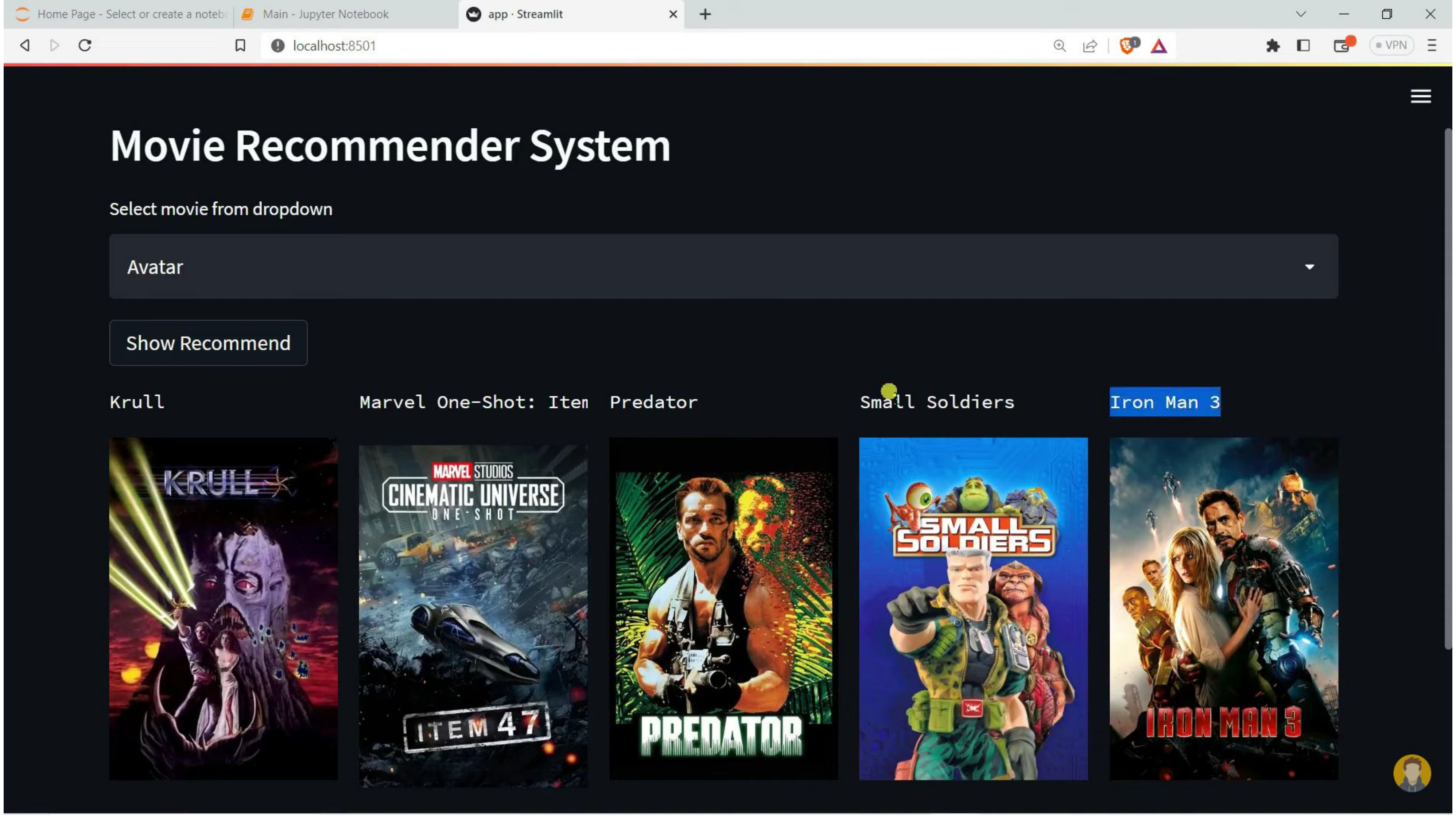
Conclusion:

Efficiently provided personalized movie recommendations.

Future Enhancements:

- Add features like cast and ratings.
- Use advanced NLP techniques.
- Scale for real-time recommendations.

RESULT



Movie Recommender System

Select movie from dropdown

Avatar

Show Recommend

Krull

Marvel One-Shot: Item

Predator

Small Soldiers

Iron Man 3

PROJECT LINK(GITHUB, GOOGLE DRIVE LINK)

Github : https://github.com/gamerGK/Netflix-movie-recommender-system

CONCLUSION

- **Summary:** Developed a functional movie recommender system using NLP and cosine similarity.
- **Effectiveness:** The system effectively recommends movies based on the content and genre.
- **Key Takeaways:** Highlighted the importance of feature engineering and similarity measures in building a recommender system.

FUTURE SCOPE

- **Enhanced Features:** Incorporate additional features like cast, crew, and keywords to improve recommendations.
- **Advanced NLP Techniques:** Utilize techniques like TF-IDF, Word2Vec, or BERT for better text understanding.
- **Real-time Recommendations:** Develop a scalable solution to provide real-time recommendations.
- **User Interaction:** Incorporate user feedback to continuously improve the recommendation system.

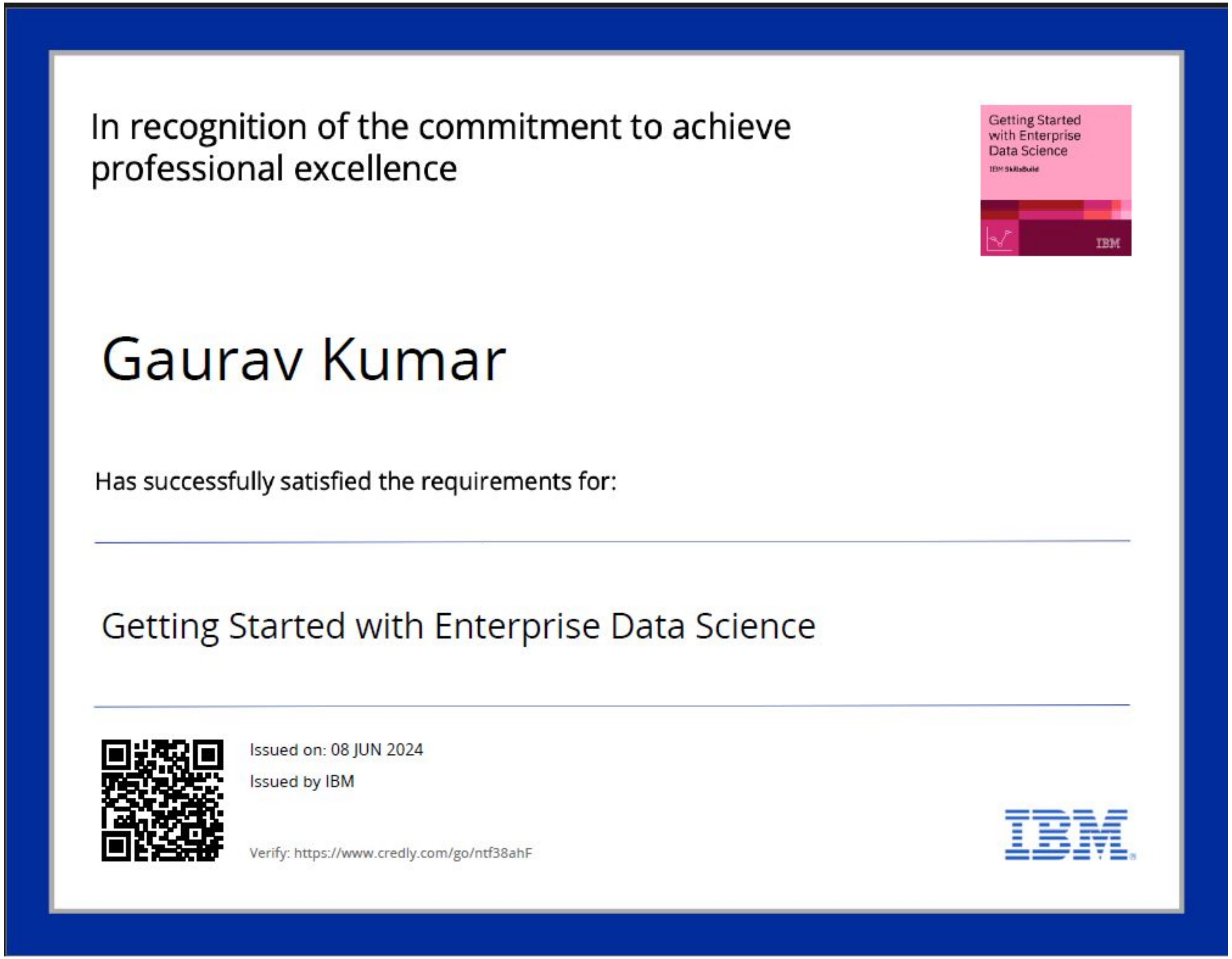
REFERENCES

Resources Used:

- **Pandas:** For data manipulation and analysis.
- **Scikit-learn:** For vectorization and similarity calculations.
- **Pickle:** For saving and loading the trained model and data.
- **Documentation:** Relevant documentation for libraries and tools used.

COURSE CERTIFICATE 1

Link:
https://www.credly.com/go/ntf38ahF



In recognition of the commitment to achieve professional excellence

Getting Started with Enterprise Data Science
IBM SkillsBuild
IBM

Gaurav Kumar

Has successfully satisfied the requirements for:

Getting Started with Enterprise Data Science

Issued on: 08 JUN 2024

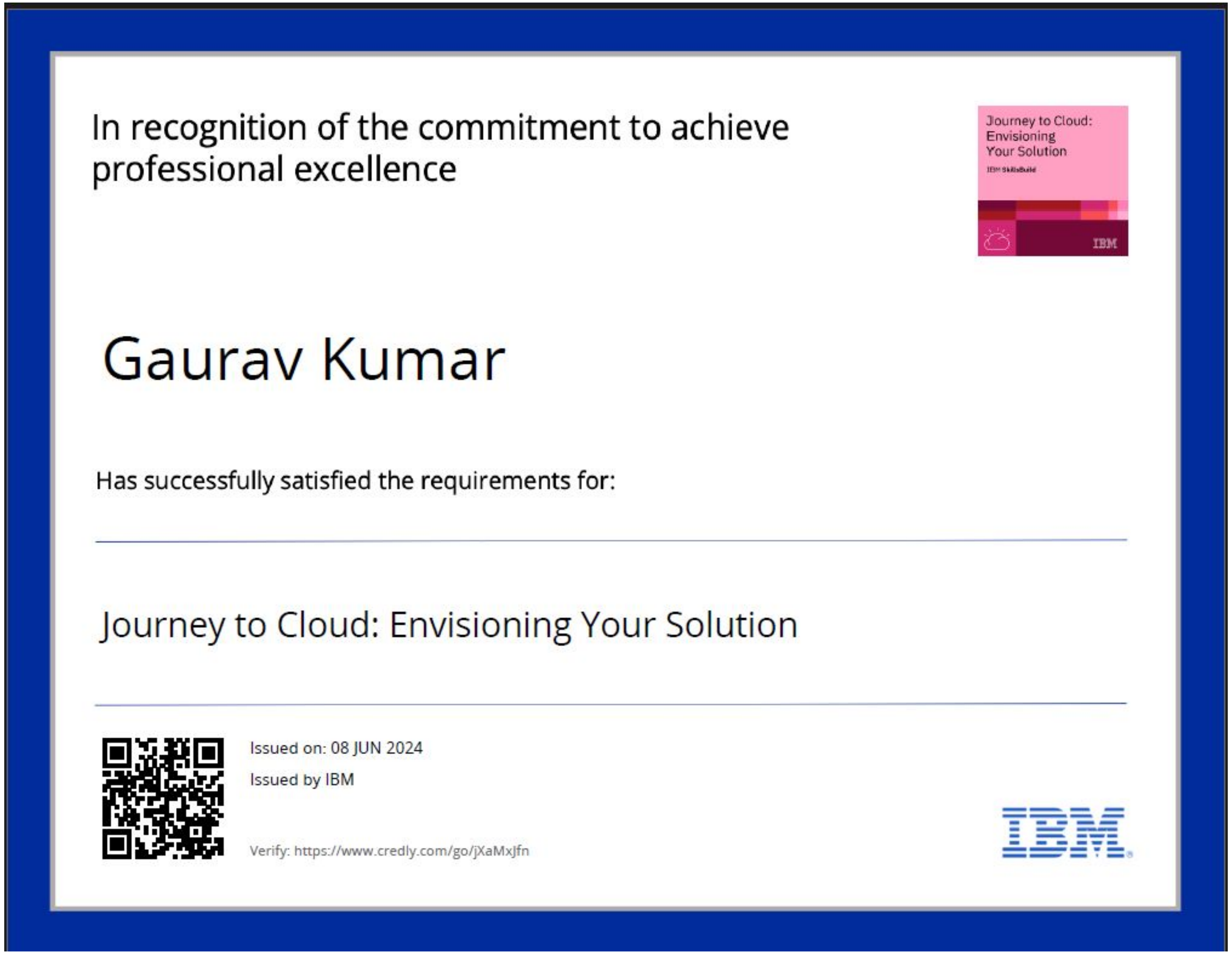
Issued by IBM

Verify: https://www.credly.com/go/ntf38ahF

IBM

COURSE CERTIFICATE 2

Link:
https://www.credly.com/go/jXaMxJfn



Journey to Cloud:
Envisioning
Your Solution
IBM SkillsBuild
IBM

In recognition of the commitment to achieve professional excellence

Gaurav Kumar

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: 08 JUN 2024

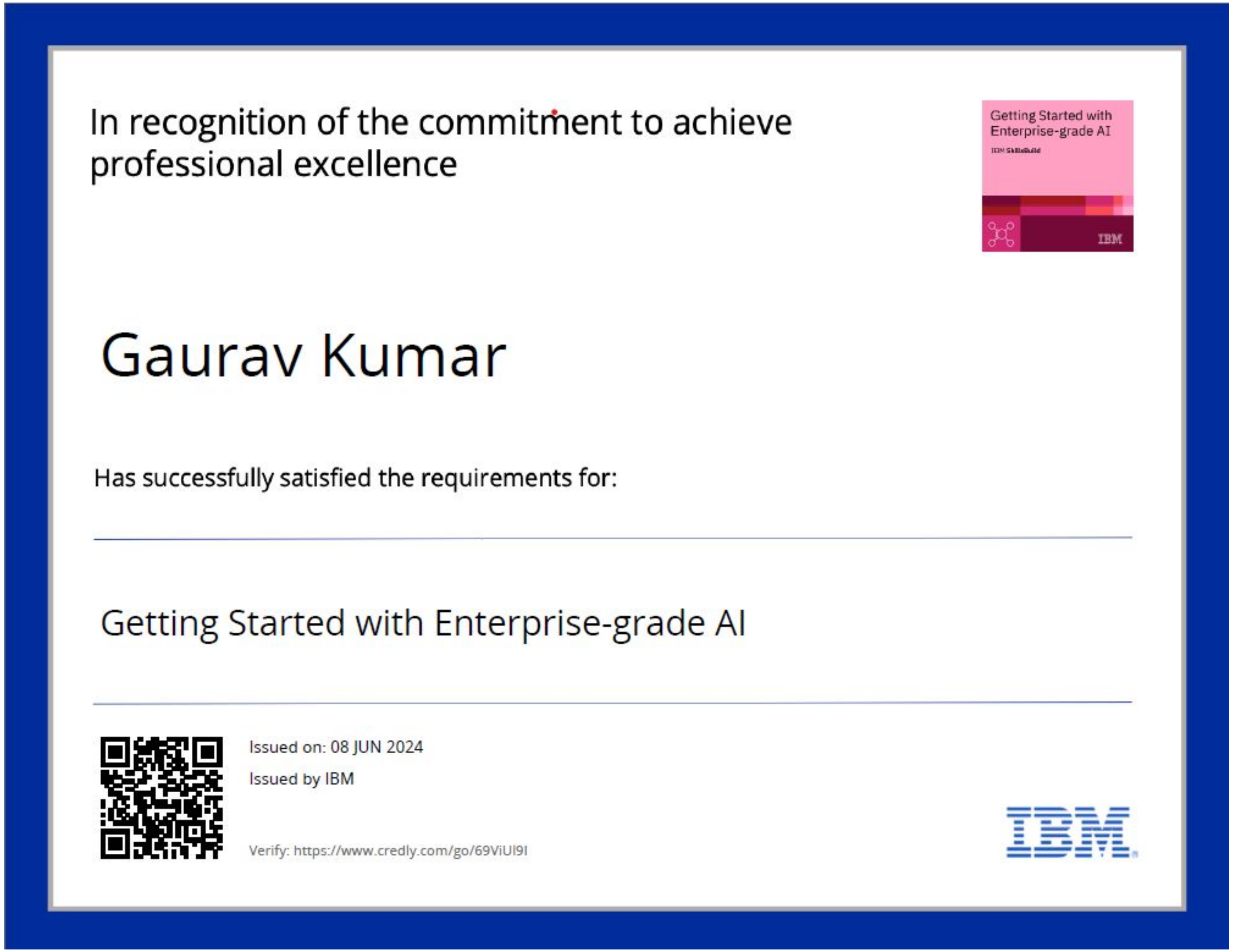
Issued by IBM

Verify: https://www.credly.com/go/jXaMxJfn

IBM

COURSE CERTIFICATE 3

Link:
https://www.credly.com/go/69ViUI9I



In recognition of the commitment to achieve professional excellence

Getting Started with Enterprise-grade AI
IBM SkillsBuild
IBM

Gaurav Kumar

Has successfully satisfied the requirements for:

Getting Started with Enterprise-grade AI

Issued on: 08 JUN 2024

Issued by IBM

Verify: https://www.credly.com/go/69ViUI9I

IBM

THANK YOU

Name: Gaurav Kumar
Student Id: INTERNSHIP_1712731032661633 98a8c87
Group no.: B1 G4
edunet Mail: gauravk8271@edunetmail.com
personal mail : gauravk8271@gmail.com
Institute name: IIT kharagpur
phone: 6204780646