Enhancing News Recommendation Systems: A Two-Tower Approach with Word2Vec Embeddings - Project Report

Veda Kailasam Sai Krishna Sanniboina Victoria Kravchuk

University of Illinois Urbana- Champaign Champaign, Illinois, USA [vedak2@illinois.edu](mailto:vedak2@illinois.edu) [ss235@illinois.edu](mailto:ss235@illinois.edu) vk16@illinois.edu

# ABSTRACT

This project proposes an advanced tweet recommendation system

to address the challenges of information overload and lack of

personalization on Twitter. The system leverages Word2Vec

embeddings to capture semantic meaning from tweets and

incorporates multi-modal features such as posting date, genre, age,

and user demographics. A Two-Tower neural network architecture

is designed to integrate these features, providing personalized and

contextually relevant tweet recommendations. The project aims to

enhance user engagement and satisfaction by delivering accurate

and diverse content tailored to individual preferences.

# KEYWORDS

Recommendation Systems, Word2Vec, Two-Tower Architecture, Preprocessing, MIND dataset,, Personalization, User Engagement, NLP.

# INTRODUCTION

The abundance of information and content on the internet has made it increasingly challenging for users to discover and consume content that aligns with their interests. Recommendation systems have emerged as a vital solution to this problem, aiming to personalize the user experience by suggesting relevant content based on individual preferences and behaviors. These systems have found extensive applications across various domains, including e-commerce, social media, and **news recommendation**.

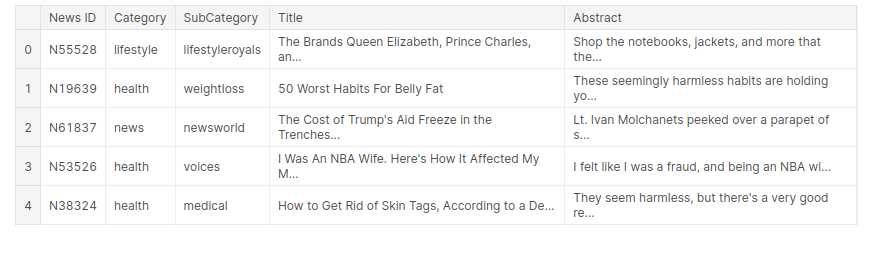
The MIND Dataset offers a rich source of data for the development and evaluation of recommendation systems, particularly in the domain of news articles recommendations. We explore the fundamental concepts and methodologies behind recommendation systems and apply them to the context of news articles. Leveraging the unique characteristics of the MIND dataset, we aim to design and evaluate a recommendation system capable of providing users with news articles that align with their preferences and interests.

We propose a two-tower-based approach for implementing a recommendation system using TensorFlow that involves designing two distinct neural networks, often referred to as "towers," to capture user and item interactions and preferences. These towers process user and item data separately and then combine their outputs to generate recommendations.

# 2 DATASET

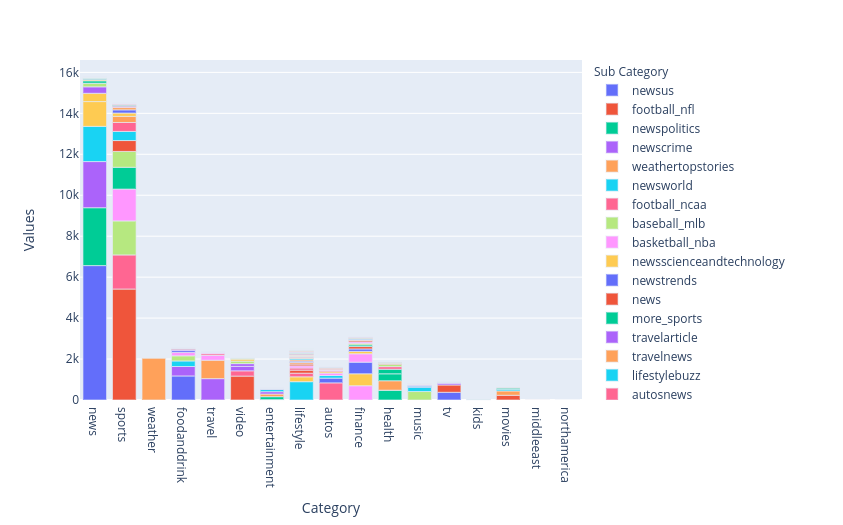
The MIND dataset, or Microsoft InnerEye NLP Dataset, is a substantial collection of news articles and user interactions, designed for various natural language processing and information retrieval tasks. It is notable for its large scale, containing millions of news articles and user engagement data like clicks and impressions. The dataset supports real-world challenges in personalization, diversity, and user intent, making it a valuable resource for NLP and recommendation system search. Key columns include News ID, Category, Subcategory, Title, Abstract, User ID, Impressions, Clicks, Dwell Time, Timestamps, and User Demographics.

* Impression ID. The ID of an impression.
* User ID. The anonymous ID of a user.
* Time. The impression time with format "MM/DD/YYYY HH:MM:SS AM/PM".
* History. The news click history (ID list of clicked news) of this user before this impression. The clicked news articles are ordered by time.
* Impressions. List of news displayed in this impression and user's click behaviors on them (1 for click and 0 for non-click). The orders of news in impressions have been shuffled.
* News ID
* Category
* SubCategory
* Title
* Abstract



# 3 PRE-PROCESSING

1. **Restructuring Data**
2. Organize the dataset for efficient analysis and modeling.
3. Arrange data into a structured format suitable for further processing.



**2. Removing Duplicates**

1. Identify and eliminate duplicate entries within the dataset to maintain data accuracy and integrity.
2. Ensure each record is unique for meaningful analysis.

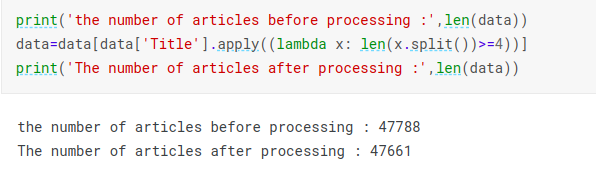
**3. Handling Null Values**

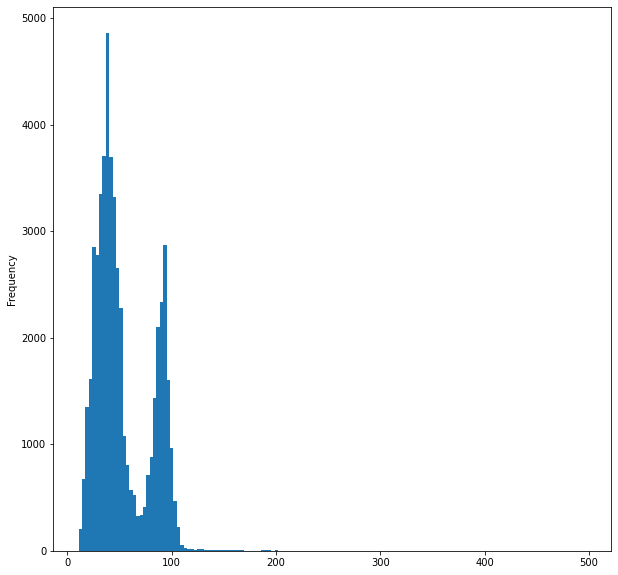
1. Address missing or null values in the dataset.
2. Implement appropriate strategies such as imputation or removal to enhance data completeness.



**4. Selecting Titles with More than 4 Words**

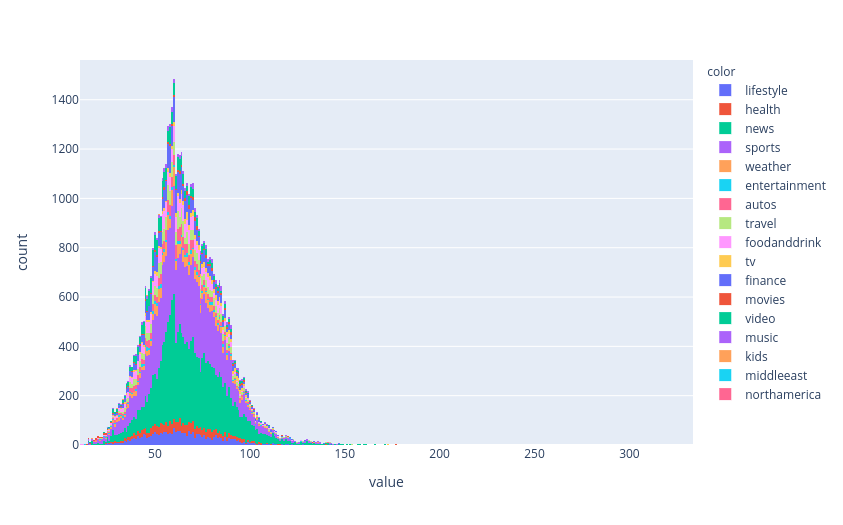
1. Filter out textual data where the title contains more than four words.
2. This criterion helps focus on detailed and descriptive entries for analysis.





**5. Document Length Normalization**

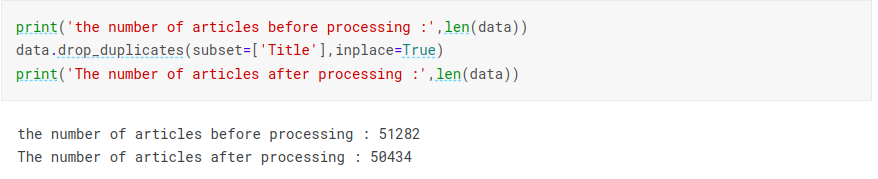
1. Normalize the length of documents or text entries to a consistent format.
2. This step ensures uniformity for accurate comparison and analysis.



**6. Text Preprocessing**

Apply essential text processing techniques like:

1. Stop Words Removal: Eliminate common words (e.g., "and," "the") to focus on meaningful content.
2. Stemming and Lemmatization: Reduce words to their base or root form, enhancing textual analysis accuracy.
3. Keywords Extraction: Identify and extract relevant keywords from the text for deeper understanding.



**7. Filtering Users and Items**

1. Users with Positive and Negative Interactions: Focus on users who have both positive and negative interactions. This selection ensures a comprehensive analysis of user behavior.
2. Items with More than 3 Interactions: Concentrate on items that have received at least three interactions. This criterion helps identify popular or significant items in the dataset.

Number of items that have less than 100 clicks make up 93.852 % of the total, and these will be removed.

Number of interactions in the behavior dataset: 781871

Number of users in the behavior dataset: 49832

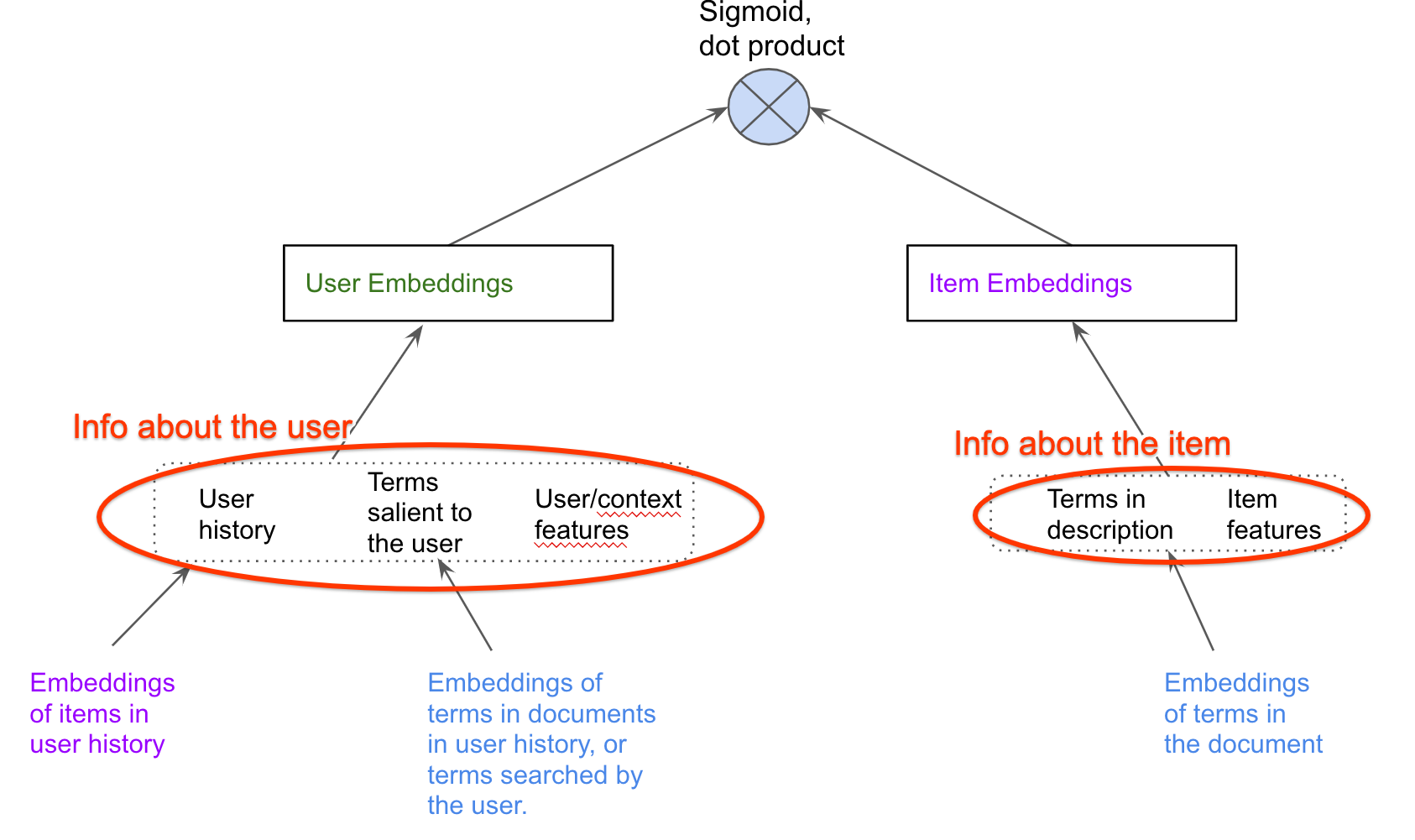
Number of articles in the behavior dataset: 2451

# 4 DESCRIPTION

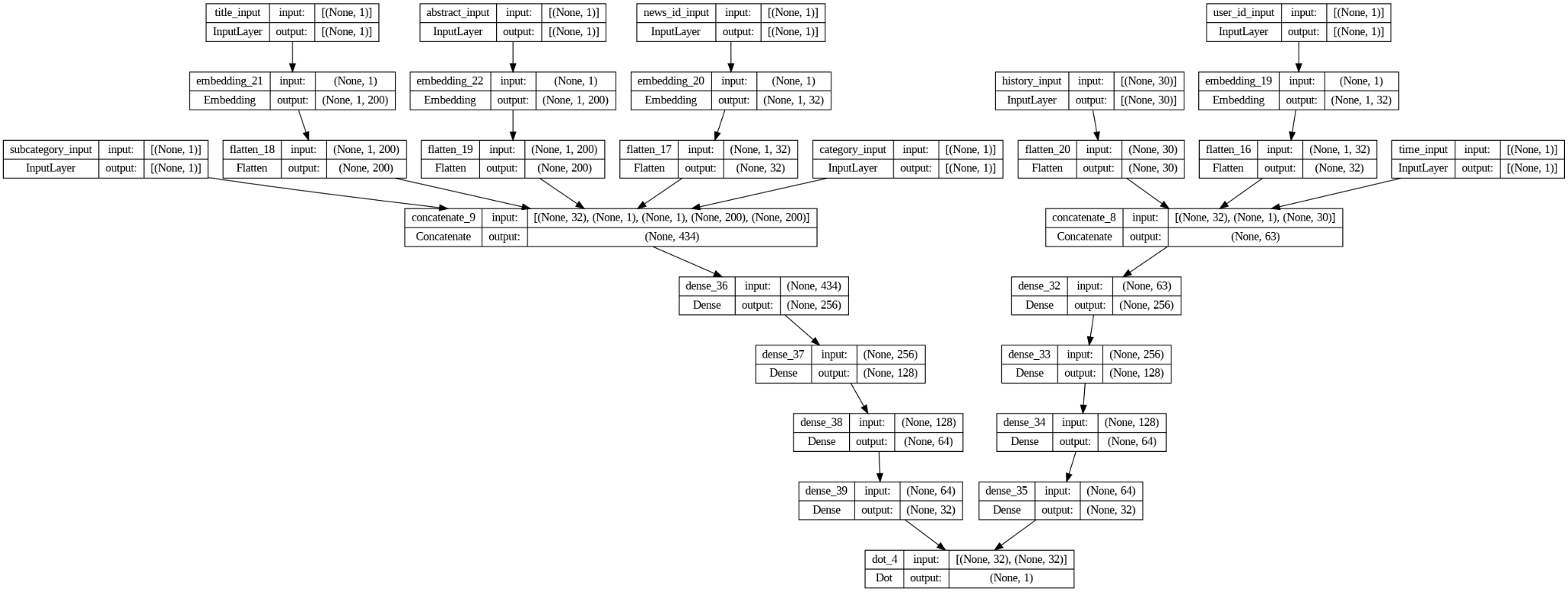
Two-Tower Neural Network Architecture is being used to generate news recommendations. This involves creation of two separate neural network branches or towers which capture the interaction between users and items where items are the news articles.

1. User Tower:
   * The user tower focuses on processing information related to the user. It typically includes embeddings or representations for user-specific features such as user IDs, historical interactions, and contextual information.
   * The tower consists of multiple layers of neural network nodes, often using activation functions like ReLU, designed to capture complex patterns and relationships in the user data.
   * The output of the user tower is a condensed representation of the user's characteristics and preferences.

* Item Tower:
  + The item tower processes information related to the items (or content) being recommended. This includes embeddings for item IDs, categorical features like category and subcategory, and textual information like titles and abstracts.
  + Similar to the user tower, the item tower consists of multiple layers of neural network nodes with activation functions to learn intricate patterns in the item data.
  + The output of the item tower is a compact representation of the item's characteristics and features.
* Interaction:
  + The outputs from the user and item towers are combined, often using operations like a dot product, element-wise multiplication, or concatenation. The goal is to capture the interactions and relationships between users and items.
  + For example, a dot product operation can represent the similarity or preference alignment between a user and an item.
* Training Objective:
  + The model is trained using a supervised learning approach, where the goal is to minimize the difference between the predicted interactions (output of the towers) and the actual interactions (ground truth labels or scores).
  + Common loss functions for recommendation systems include mean squared error for regression tasks or binary cross-entropy for binary classification tasks (e.g., like/dislike).



*Two-Tower Interaction*



# *Model Architecture*

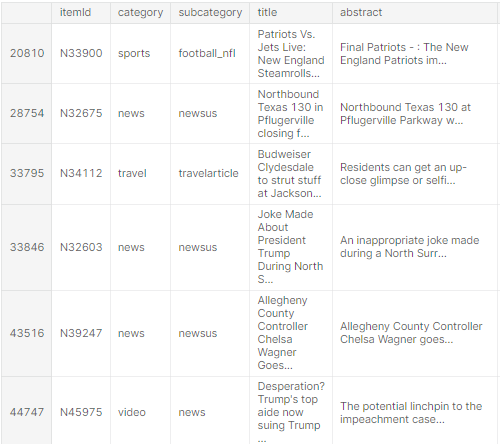
**5 RESULTS**

### Recall@50:

### Recall is a measure that evaluates the ability of the recommender system to capture relevant items in the top-k recommendations. In this context, Recall@50 means that 68% of the relevant items were successfully included in the top 50 recommendations made by your system. A high Recall@50 value implies that your recommender system is adept at identifying and suggesting items that the user is likely to be interested in, and it is effective in not missing a significant portion of relevant items.

### Hit Ratio@50:

### Hit Ratio@k measures the proportion of user interactions where at least one relevant item is present in the top-k recommendations. In this case, a Hit Ratio@50 of 33% indicates that, for 33% of user interactions, there is at least one relevant item among the top 50 recommendations made by your system.



***History***

******

***Recommended***

**6 DISCUSSION**

In conclusion, the two-tower architecture employed in our recommender system has demonstrated a commendable performance, achieving a fine balance between Recall@50 and Hit Ratio@50. This equilibrium is indicative of the system's proficiency in identifying relevant items (Recall) and consistently including them in the top recommendations (Hit Ratio), thereby enhancing the overall user experience. The observed positive trade-off underscores the system's capability to cater to diverse user preferences effectively.

Potential enhancements could involve the implementation of negative sampling to enhance model robustness. Negative sampling offers an avenue to address challenges and further refine the system's recommendations. Additionally, the inclusion of both implicit and explicit multi-target labels, such as likes and watch time, holds promise for refining and personalizing recommendations. These enhancements not only contribute to the system's accuracy but also ensure adaptability to evolving user behaviors and preferences. In summary, the two-tower architecture has laid a strong foundation, setting the stage for future improvements and innovations that will continue to elevate the recommender system's performance and user satisfaction.