A Comparative Analysis of TF-IDF and Skip-Gram Word Embeddings for QA Text Classification using Deep Learning Architectures

Amir Sakib Saad
Department of Computer Science and Engineering
BRAC University, Dhaka
Dhaka, Bangladesh
amir.sakib.saad@g.bracu.ac.bd

Asef Ahmed Shimanto
Department of Computer Science and Engineering
BRAC University, Dhaka
Dhaka, Bangladesh
asef.ahmed.shimanto@g.bracu.ac.bd

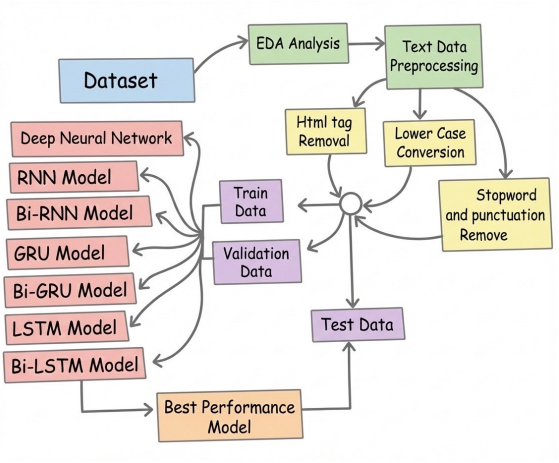
Faiyaz Zaman Saadman
Department of Computer Science and Engineering
BRAC University, Dhaka
Dhaka, Bangladesh
shah.faiyaz.zaman@g.bracu.ac.bd

***Abstract*—In this venture, diverse human language handling (NLP) approaches for automated script sorting within a question-and-answer (QA) structure are comprehensively weighed. The efficiency of two distinct content description strategies is examined in this endeavor: prediction-based Word2Vec (Skip-Gram) embeddings and frequency-based TF-IDF (Term Frequency-Inverse Document Frequency). Numerous machine learning and deep learning designs, including multinomial naive Bayesian, deep neural networks (DNN), and various recurrent neural networks (RNN) like simpleRNN, long short-term memory (LSTM), and gated recurrent units (GRU), were combined with these vectorization schemes. Sentiment charting and n-gram illustration are employed in exploratory data scrutiny (EDA) after a meticulous prose preparation phase that incorporates stop word straining, HTML tag extraction, and lowercase conversion. To establish how chronological setting influenced classification precision, performance was appraised on both unidirectional and bidirectional variations of the recurrent models. Confusion matrices and exhaustive classification summaries are used to display experimental figures and demonstrate how computational intricacy and forecasted accuracy are linked. In order to boost the consistency of automated quality assurance categorization setups, this inquiry offers perspectives on the ideal word embedding pairings and model layouts.**

***Index Terms*—RNN, LSTM, GRU, NLP, DNN, EDA, TF-IDF**

I. INTRODUCTION

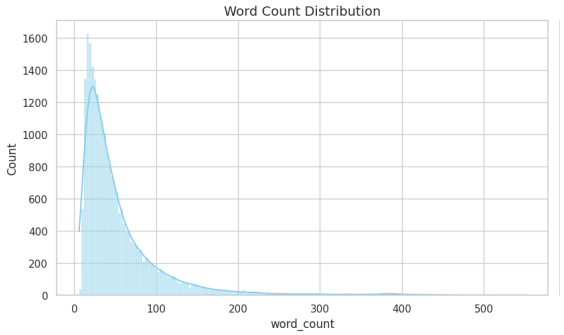
The relationship between artificial intelligence (AI) and healthcare has emerged as a key area for enhancing information accessibility and accuracy in the current digital era. Medical QA (Question and Answer) systems, which offer prompt responses to complicated queries, are vital resources for both the general public and healthcare professionals. The creation and comparison of several natural language processing (NLP) models intended to efficiently categorize medical literature is the main goal of this research. This study aims to determine the best reliable techniques for comprehending and categorizing medical conversation using a range of structured approaches, from conventional statistical techniques to sophisticated deep learning frameworks. The meticulous data preparation procedure is the project's cornerstone. Words like HTML tags, punctuation, and common stop words that don't add to the semantic meaning of clinical questions are frequently seen in raw medical language. The project addresses this by implementing a custom cleanup function that eliminates non-alphabetic letters, filters out non-informative linguistic parts, and normalizes the text to lowercase. This stage is critical because the quality and cleanliness of the input sequence have a significant impact on the performance of complicated neural networks like supervised recurrent units (GRU) and long-term memory (LSTM). Feature engineering and text representation have been extensively studied. Word2Vec (Skip-gram) and TF-IDF (Inverse Term Frequency-Document Frequency) are the two primary methods that have been compared. Skip-Gram is one of the two architectures proposed in Word2Vec for learning distributed word representations by predicting context words from a center word.



[4]The statistical significance of words in a corpus is extracted using TF-IDF, which serves as the foundation for conventional machine learning models such dense neural networks and basic Bayesian polynomials. On the other hand, dense vector embeddings are produced using Word2Vec's Skip-gram implementation. In medical language, where the closeness and link between terms like "symptom," "diagnosis," and "treatment" are crucial, these embeddings enable models to capture semantic and contextual relationships. Lastly, the project uses descriptive performance measures to highlight empirical validation. The study used descriptive confusion matrices, precision-recall curves, and F1 scores in addition to basic accuracy to assess the models' performance. This study offers a thorough understanding of how vectorization strategies impact the outcomes by contrasting a TF-IDF-based dense network with a Skip-Gram-based deep neural network (DNN). In the end, our effort advances the overarching objective of creating more intelligent, pertinent systems that can comprehend the subtleties of medical terminology to deliver trustworthy information.

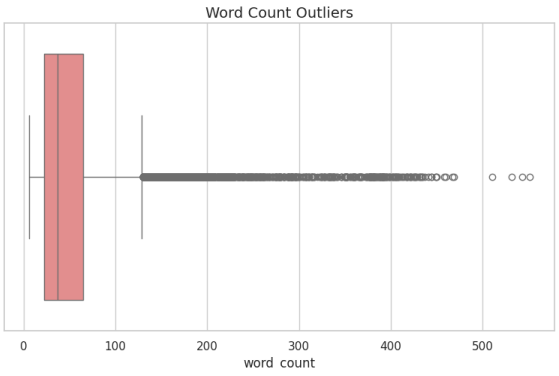
II. METHODOLOGY

Exploratory data analysis, normally known as arsenic EDA, is an important initial measure in data analysis, where the basic structure, form, and features of the data are fully understood. This measure measures the accuracy, completeness, and suitability of the data for analysis rather than building an angstrom model from the data. EDA helps identify whether the dataset has missing hour angle values, outliers, or abnormal values. It helps determine the distribution of the data. In addition, the correlation between different features is analyzed, which is necessary for the approaching feature choice and model development. Descriptive statistics and assorted visual image techniques, such as arsenic scatter plots, box plots, and histograms, are normally used to complete EDA. When EDA is performed properly, there are fewer misconceptions about the data, and as a consequence, machine learning or deep learning models can supply more accurate consequences.



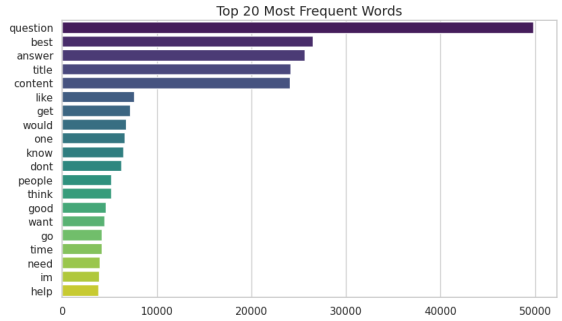
This distribution is strongly right-skewed. Positively skews distribution. This means that most of the data values in the dataset are concentrated on the left side, i.e., towards lower berth values. We can see that the number of words in most texts or documents is expressed as 20 to 60. The high extreme, 'Peak,' of the graph bespeaks that the number of texts with an angstrom length of 20-25 words is high (about 1600 More). This is named the manner in which statistics.

The box plot entitled word count outlier exemplifies an angstrom distribution that is heavily skewed to the right, bespeaking that the majority of the data points are concentrated at lower berth values, while a significant number of extreme observations pull the tail toward the higher end. The central box, which represents the interquartile scope where the center 50% of the data reside, is positioned approximately between 20 and 70 words, with the angstrom median line appearance approaching the 40-word mark. The box plot titled "Word



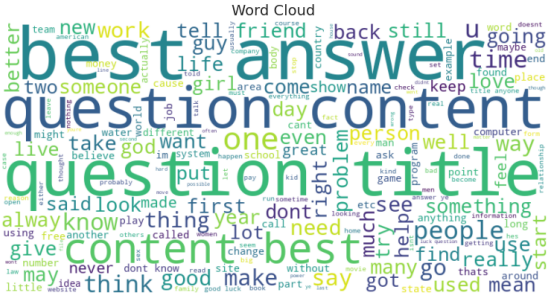
Count Outliers" illustrates a distribution that is heavily skewed to the right, indicating that the majority of the data points are concentrated at lower values while a significant number of extreme observations pull the tail toward the higher end. The central box, which represents the interquartile range where the middle 50% of the

data resides, is positioned roughly between 20 and 70 words, with a median line appearing near the 40-word mark. The relatively short left whisker suggests a firm floor for the minimum word count, whereas the right whisker extends to approximately 130 words before transitioning into a dense and persistent trail of outliers. These outliers, depicted as individual circles, extend far beyond the typical range to reach a maximum of over 550 words. This visual pattern reveals that while the dataset primarily consists of short-form content, it also contains a substantial volume of ”long-tail” entries that are statistically distinct from the core population and may require specialized handling, such as truncation or normalization, during data preprocessing.



The horizontal bar chart titled ”Top 20 Most Frequent Words” illustrates the dominance of specific terms within a text dataset, revealing a clear hierarchical structure where a small group of words occurs significantly more often than the rest. The word ”question” stands as the most prominent outlier, appearing approximately 50,000 times, which is nearly double the frequency of the next most common words, ”best” and ”answer,” both of which hover around the 25,000 to 27,000 range alongside ”title” and ”content.” This initial cluster of five words suggests the dataset is likely sourced from a Q&A platform or an inquiry-based community where structural elements of a post are frequently mentioned. Following this top tier, there is a sharp drop in frequency to a secondary group of words like ”like,” ”get,” and ”would,” which appear fewer than 10,000 times each. As the list descends toward ”help” at the bottom of the top 20, the frequencies become much more uniform and stable, consisting largely of common verbs and pronouns that characterize natural conversational language. The color gradient—shifting from deep purple for high-frequency terms to bright yellow for lower ones—further emphasizes this steep decline in word usage, highlighting how a few specialized terms define the core thematic focus of the entire corpus.

The provided word cloud serves as a visual representation of text data, where the physical size of each word directly correlates with its frequency or importance

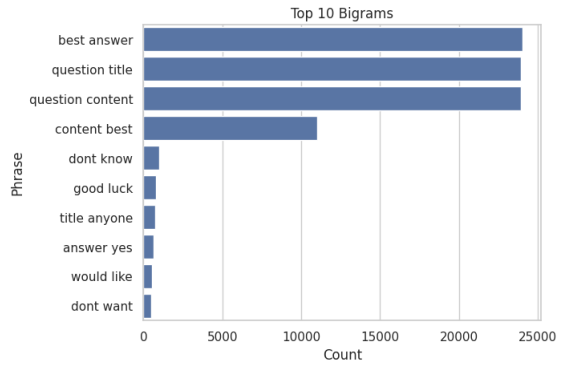


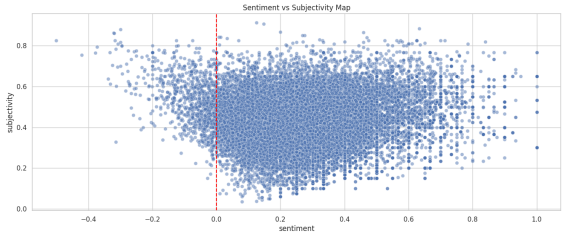
within the underlying dataset. Dominant terms such as question, answer, content, title, and best are prominently displayed in the largest fonts, suggesting that the source material likely originates from an interactive information-sharing environment like a QA forum, a customer support database, or a knowledge management system. The high density of words like ”people,” ”friend,” ”life,” and ”know” points toward a focus on human interaction and social exchange, while secondary terms like ”work,” ”time,” ”year,” and ”need” add a layer of practical or task-oriented context. By consolidating these keywords into a single visualization, the cloud highlights a discourse centered on seeking solutions, evaluating the quality of information, and addressing everyday inquiries or personal experiences. The lack of highly specialized technical jargon suggests the content is general-purpose and accessible, emphasizing utility and collaborative problem-solving through the repetition of verbs like ”make,” ”think,” and ”find.”

This horizontal bar chart, titled ”Top 10 Bigrams,” illustrates the frequency of two-word combinations within a dataset, highlighting a strong focus on information exchange and content structure. The three most frequent bigrams—best answer, question title, and question content—all share a nearly identical, dominant count exceeding 20,000 occurrences, which strongly suggests that the data is sourced from a structured QA platform or a technical support forum where users categorize their posts into titles and bodies while seeking validated solutions. These are followed by ”content best,” which appears approximately half as often, likely indicating a recurring theme of evaluating the quality of information.

The remaining bigrams, such as ”dont know,” ”good luck,” ”answer yes,” and ”would like,” appear with significantly lower frequency, representing common conversational fillers or expressions of intent and politeness within the user interactions. Overall, the significant gap between the top four phrases and the rest of the list underscores a highly standardized language pattern revolving around the formal components of a question-and-answer ecosystem.

This scatter plot, titled Sentiment vs Subjectivity Map,





visualizes the emotional tone and personal nature of a dataset by plotting individual data points along two axes. The horizontal axis represents sentiment, ranging from -0.5 (negative) to 1.0 (positive), while the vertical axis measures subjectivity from 0.0 (objective/factual) to nearly 1.0 (subjective/opinion-based). A clear observation is the high density of points concentrated to the right of the red dashed line at 0.0, indicating that the overall sentiment of the discourse is predominantly positive. Furthermore, the bulk of the data is clustered between 0.3 and 0.7 on the subjectivity scale, suggesting that the content is largely composed of personal opinions, experiences, or qualitative assessments rather than purely clinical or objective facts. The wide spread of points, particularly the ”fanning out” effect as sentiment becomes more positive, implies that users express a diverse range of subjective views when they are happy or satisfied, whereas negative sentiments appear slightly more sparse and less varied in their subjectivity. This distribution reinforces the idea of a social or community-driven platform where helpfulness and positive engagement are the norm, aligning with the ”best answer” and ”good luck” themes seen in previous charts.

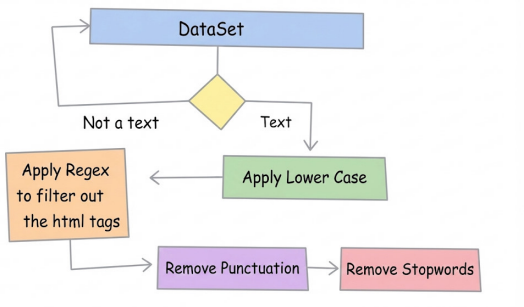
	QA Text	Class	char_count	word_count	sentiment	subjectivity
0	question title el chocolate es lo mas rico per...	Society & Culture	190	32	0.297917	0.514583
1	question title inlaws called momdad first name...	Family & Relationships	339	52	0.360417	0.406250
2	question title personals yahoo page colored pi...	Business & Finance	127	20	0.287500	0.434722
3	question title many african leaders foreign ac...	Politics & Government	174	25	0.186111	0.413889
4	question title get burned letter scar show way...	Health	260	40	0.606667	0.581111
...	...	...	...	...	...	...
23906	question title download driver bt voger 105 ma...	Computers & Internet	220	24	0.700000	0.566667
23907	question title guys correct etiquette deal que...	Society & Culture	461	67	0.137500	0.368750
23908	question title communicate boyfriends mom ques...	Family & Relationships	554	87	0.211111	0.562054
23909	question title language would like learn quest...	Education & Reference	336	48	0.144048	0.197024
23910	question title find wwwquikpaydaycom web quest...	Business & Finance	235	40	0.625000	0.625000

Sentiment (Polarity) Sentiment analysis usually measures ”polarity”—whether a piece of text is positive, negative, or neutral.

The algorithm compares the words in your QA Text against a pre-defined dictionary where words have assigned scores. For example, ”excellent” might be +0.8, while ”terrible” is 0.8. The Formula: The final score is often the average of the scores of all sentiment-bearing words in the string. It ranges from -1.0 (very negative) to 1.0 (very positive). A score of 0 is neutral. To understand if users are frustrated (negative sentiment in ”Health” or ”Computers”) or happy. To see if certain topics (e.g., ”Politics Government”) trigger more hostile interactions compared to others. Subjectivity quantifies the amount of personal opinion, emotion, or judgment versus factual information in the text. the algorithm looks for ”subjective words” (adjectives like beautiful, annoying, great) versus ”objective words” (nouns and factual verbs). It ranges from 0.0 to 1.0. 0.0: Very objective (factual, like a news report). 1.0: Very subjective (pure opinion or emotion).In categories like ”Education Reference,” you might want low subjectivity (facts). In ”Family Relationships,” you expect high subjectivity (personal stories).It helps distinguish between a user asking for a technical fix (objective) and a user venting about a bad experience (subjective).

Textual data preprocessing is the process of cleaning and ”standardizing” raw text so that machine learning algorithms can understand it. Because computers are great at math but poor at understanding language nuances, we must transform messy, human-written text into a structured format. However there can be diverse way to preprocessed the textual data. But here we use a customized pipeline that handles the text of the focused dataset.

NLP Preprocessing Workflow Formalization



The function $f(\text{text})$ transforms a raw input string into a cleaned format through a series of mathematical and logical operations.

1. Input Validation

The function first ensures the input x is a member of the set of strings $\mathbb{S}$. If $x \notin \mathbb{S}$, the function returns the identity:

$$f(x) = \begin{cases} \text{Proceed to Step 2} & \text{if } x \in \mathbb{S} \\ x & \text{if } x \notin \mathbb{S} \end{cases}$$

2. Normalization and Noise Removal

Let T_{raw} be the input string. We define the following operations:

- **Case Folding:** $T_1 = \{c.\text{lower}() \mid \forall c \in T_{raw}\}$
- **Regex Cleaning:** $T_2 = \text{sub}(\text{pattern} = \langle [^{>}]+ \rangle, \text{repl} = \epsilon, T_1)$
- **Whitespace Handling:** $T_3 = \text{replace}(\text{'n'}, \text{' '}, T_2)$

3. Punctuation Stripping

Let P be the set of all punctuation characters defined in `string.punctuation`. The transformation is:

$$T_4 = \{c \mid c \in T_3, c \notin P\}$$

4. Tokenization and Stop Word Filtering

The string is split into a sequence of tokens $W = \{w_1, w_2, \ldots, w_n\}$. Given a set of stop words S, the filtered sequence W_{clean} is defined as:

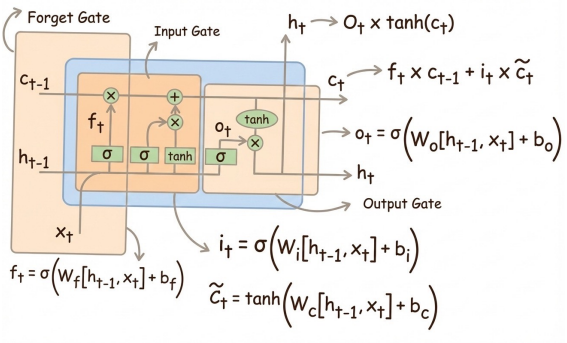
$$W_{clean} = \{w_i \in W \mid w_i \notin S\}$$

5. Re-joining

The final output T_{final} is the concatenation ($\oplus$) of the elements in W_{clean} separated by a space character σ:

$$T_{final} = \bigoplus_{i=1}^{|W_{clean}|} (w_i + \sigma)$$

MODEL A: LSTM ARCHITECTURE



1) **Forget Gate (f_t):** Decides what to remove from the cell state.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

2) **Input Gate (i_t):** Decides what new information to store.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

Candidate Update:

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

3) **Cell State Update:** The old state is multiplied by the forget factor, and the new candidate memory is added.

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

4) **Output Gate (o_t):** Decides what to output as the hidden state based on the filtered cell state.

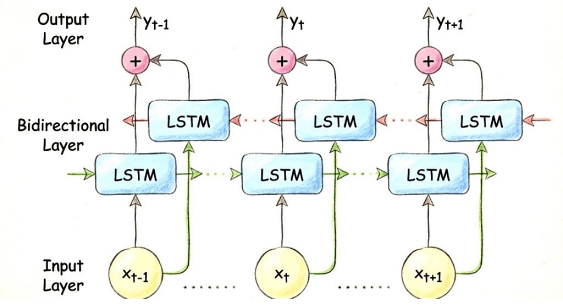
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

Final Hidden State:

$$h_t = o_t * \tanh(C_t)$$

LSTM is an enhanced RNN designed to mitigate the vanishing gradient problem and preserve long-term dependencies in sequences. It does this via memory cells and gates (input, forget, output). [2]

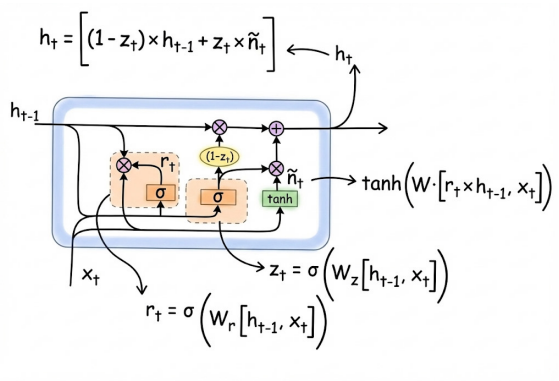
MODEL B: BI-DIRECTIONAL LSTM ARCHITECTURE



- **Structure:** It utilizes two parallel LSTM tracks: one processing the sequence forward and one backward.
- **Dataflow:** The input x_t is sent to both LSTMs. The final hidden state h_t is the concatenation of the forward LSTM output $\vec{h}_t$ and backward LSTM output $\overleftarrow{h}_t$:

$$h_t = [\vec{h}_t \parallel \overleftarrow{h}_t]$$

- **Key Advantage:** It captures context from both the past and the future, making it highly effective for complex NLP tasks prior to the rise of Transformers.



MODEL C: GRU ARCHITECTURE

1) **Reset Gate (r_t):** Determines how much of the past information to forget.

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t])$$

2) **Update Gate (z_t):** Determines how much of the past information to pass along to the future.

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t])$$

3) **Current Memory Content ($\tilde{h}_t$):**

$$\tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t])$$

4) **Final State Update:**

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

MODEL D: BI-DIRECTIONAL GRU ARCHITECTURE

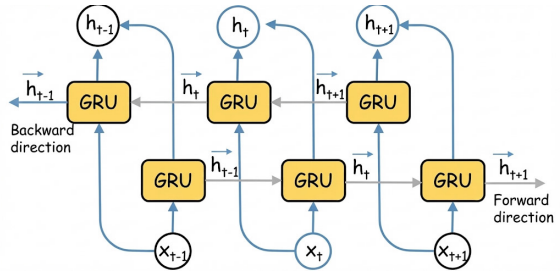
- **Concept:** Applies the bidirectional processing strategy to GRU cells to capture dependencies from both past and future time steps.
- **Architecture:**
  - **Structure:** Two parallel GRU layers operating in opposite directions.
  - **Dataflow:** The forward hidden state $\vec{h}_t$ and backward hidden state $\overleftarrow{h}_t$ are updated independently and then concatenated:

$$\vec{h}_t = \text{GRU}_{fwd}(x_t, \vec{h}_{t-1})$$

$$\overleftarrow{h}_t = \text{GRU}_{bwd}(x_t, \overleftarrow{h}_{t+1})$$

$$h_t = [\vec{h}_t \parallel \overleftarrow{h}_t]$$

- **Performance:** It offers a balance between the high performance of Bi-LSTMs and the computational efficiency of standard RNNs. It is often faster to train than Bi-LSTM with comparable accuracy for smaller datasets.



MODEL E: RNN ARCHITECTURE

Step-by-Step Data Processing

1) **Input:** At time t, the network receives input x_t and the hidden state from the previous step h_{t-1}.
2) **Combine:** These two vectors are concatenated or combined using matrix multiplication.
3) **Activation:** The combined data is passed through an activation function (usually $\tanh$) to squash values between -1 and 1.
4) **Output:** This produces the new hidden state h_t, which serves as the memory for the next step and (optionally) produces an output y_t.

RNNs are neural networks with cyclic connections that make them suitable for sequence data like text, speech, and time series. They are foundational architectures in many NLP tasks because they can "remember" previous inputs in a sequence. [1]

Mathematical Formulation

- **Update Rule:**

$$h_t = \tanh(W_h \cdot h_{t-1} + W_x \cdot x_t + b)$$

- **Output Rule:**

$$y_t = W_y \cdot h_t + b_y$$

(Where W are weight matrices and b are bias vectors)

Limitations

- **Vanishing Gradient:** As sequences get longer (e.g., > 10 steps), the gradients used for learning shrink exponentially, making it impossible for the network to learn dependencies between distant time steps.

MODEL E: BI-DIRECTIONAL RNN ARCHITECTURE

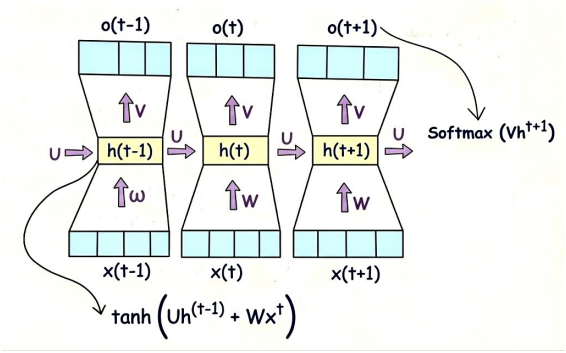
Step-by-Step Data Processing

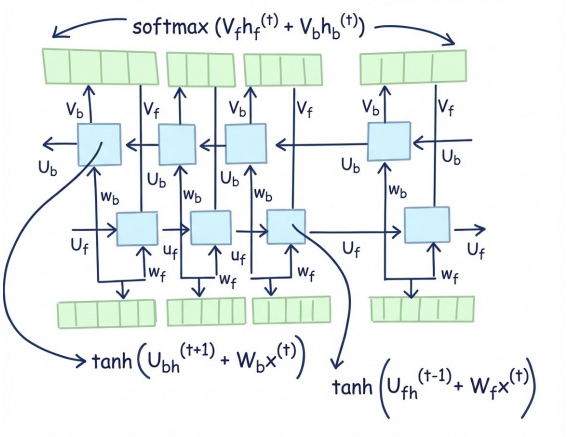
1) **Forward Pass:** One RNN layer calculates hidden states from $t = 1$ to $t = N$.

$$\vec{h}_t = \sigma(W_{\vec{h}} x_t + V_{\vec{h}} \vec{h}_{t-1} + b_{\vec{h}})$$

2) **Backward Pass:** A second RNN layer calculates hidden states from $t = N$ to $t = 1$.

$$\overleftarrow{h}_t = \sigma(W_{\overleftarrow{h}} x_t + V_{\overleftarrow{h}} \overleftarrow{h}_{t+1} + b_{\overleftarrow{h}})$$

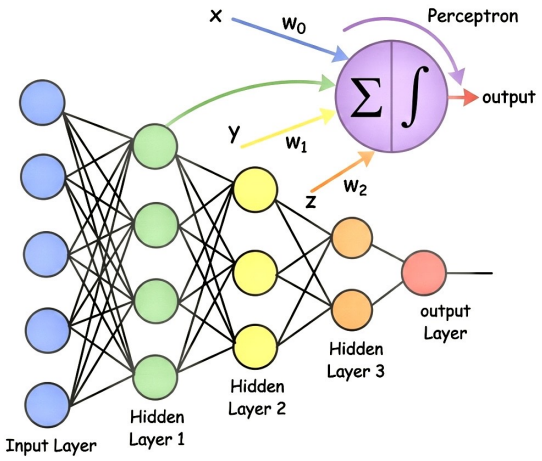




3) **Integration:** At every time step t, the final output is generated by combining the hidden state from the forward layer ($\vec{h}_t$) and the backward layer ($\overleftarrow{h}_t$).

$$y_t = g(W_y[\vec{h}_t \parallel \overleftarrow{h}_t] + b_y)$$

MODEL E: DEEP NEURAL NETWORK ARCHITECTURE



Plaintext Structure

[Input] → [Hidden 1] → [Hidden 2] → ⋯ → [Output]

$$(x) \to (h_1) \to (h_2) \to (y)$$

DNNs are feedforward neural networks with multiple hidden layers that can learn hierarchical features, widely used across vision, speech, and NLP. RNN/LSTM are special cases when applied to sequence tasks. [3]

Detailed Dataflow

1) **Input:** Features are fed into the input layer as a flat vector x.
2) **Weighted Sum:** Each neuron in Layer 1 calculates a weighted sum of inputs plus a bias:

$$z = Wx + b$$

3) **Activation:** The sum is passed through a non-linear activation function (e.g., ReLU, Sigmoid):

$$a = \sigma(z)$$

4) **Propagation:** This output a becomes the input for the next layer. This process repeats until reaching the final output layer.

III. RESULT

A. Comparative Analysis

At a high level, the models are strikingly similar. Both achieve an identical accuracy, macro average, and weighted average of 0.80. However, distinct behaviors emerge at the category level as summarized in Table X.

TABLE I
MODEL COMPARISON BY CATEGORY (TRANSPOSED)

Category	Bi-LSTM ($F1$)	Uni-LSTM ($F1$)	Winner
Comp. & Int.	0.90	**0.91**	Uni-Dir.
Bus. & Fin.	0.61 (Rec)	**0.67**	Uni-Dir.
Pol. & Govt.	**0.90 (Rec)**	0.88 (Pre)	Tie
Edu. & Ref.	0.76 (Pre)	**0.70**	Uni-Dir.
Sports	**0.89**	0.87	Bi-Dir.

B. Precision vs. Recall in Politics

- **Bi-Directional LSTM:** Favors Recall (0.90). It is aggressive in identifying political text, catching most instances but triggering more false alarms (lower precision).
- **Uni-Directional LSTM:** Favors Precision (0.88). It is more conservative; while it misses some political texts (Recall 0.83), it is highly accurate when it does flag them.

REPORT OF BI-DIRECTIONAL LSTM

	precision	recall	f1-score	support
Business & Finance	0.71	0.61	0.66	1884.0
Computers & Internet	0.89	0.91	0.90	1883.0
Education & Reference	0.76	0.81	0.67	1884.0
Entertainment & Music	0.76	0.81	0.79	1884.0
Family & Relationships	0.79	0.83	0.81	1884.0
Health	0.83	0.86	0.65	1884.0
Politics & Government	0.79	0.90	0.85	1884.0
Science & Mathematics	0.80	0.82	0.81	1884.0
Society & Culture	0.75	0.72	0.73	1883.0
Sports	0.87	0.92	0.89	1884.0
accuracy	0.80	0.80	0.80	0.8
macro avg	0.80	0.80	0.79	18838.0
weighted avg	0.80	0.80	0.79	18838.0

REPORT ONI -DIRECTIONAL LSTM

	precision	recall	f1-score	support
Business & Finance	0.71	0.61	0.66	1884.0
Computers & Internet	0.89	0.91	0.90	1883.0
Education & Reference	0.76	0.81	0.67	1884.0
Entertainment & Music	0.76	0.81	0.79	1884.0
Family & Relationships	0.79	0.83	0.81	1884.0
Health	0.83	0.86	0.65	1884.0
Politics & Government	0.79	0.90	0.85	1884.0
Science & Mathematics	0.80	0.82	0.81	1884.0
Society & Culture	0.75	0.72	0.73	1883.0
Sports	0.87	0.92	0.89	1884.0
accuracy	0.80	0.80	0.80	0.8
macro avg	0.80	0.80	0.79	18838.0
weighted avg	0.80	0.80	0.79	18838.0

C. Context vs. Complexity

- **Bi-Directional:** Theoretically superior as it processes text forwards and backwards for full context. This explains its success in *Sports*, where context changes rapidly. However, it is computationally $\approx 2\times$ more expensive to train and deploy.
- **Uni-Directional:** Processes text linearly. Despite this, it matches the Bi-Directional model's overall score, suggesting backward context was not strictly necessary for this specific dataset.

While the decision is close, the Uni-Directional LSTM is the superior choice for this specific application based on:

1) **Efficiency:** It achieves the same overall accuracy (80%) with significantly lower computational cost and faster inference.
2) **Stability:** It offers better F1-score stability in difficult categories like *Business & Finance* and *Education*.

a) Recommendation:: Use the Bi-Directional model only if the application requires maximum sensitivity (Recall) for Sports or Politics and computational resources are abundant. Otherwise, the Uni-Directional model provides the best return on investment.

IV. Overview of the Confusion Matrix

A confusion matrix provides a visual representation of model performance.

- **Y-Axis (Actual):** Represents the true categories of the text.
- **X-Axis (Predicted):** Represents the model's classification.
- **The Diagonal:** Indicates correct predictions; higher values here signify better performance.
- **Off-Diagonal:** Indicates errors or "confusions" between classes.

V. Comparative Analysis

A. Uni-Directional Strengths: Business & Education

The Uni-Directional model excels at distinguishing factual, "dry" topics:

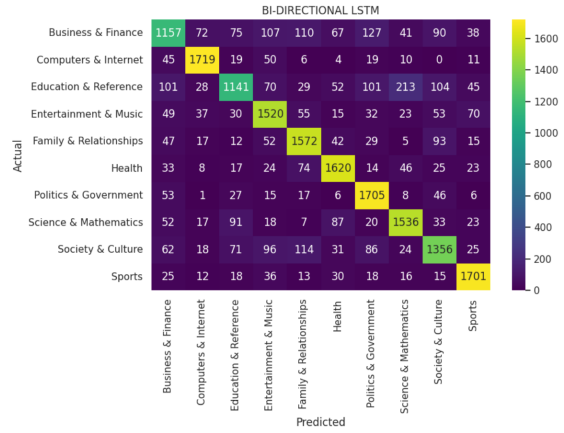
- **Business & Finance:** The Uni-Directional model correctly identified 1,313 samples, whereas the Bi-Directional model managed only 1,157. The Bi-Directional model frequently confused this category with Politics and Society.
- **Education & Reference:** The Uni-Directional model was much cleaner (1,307 correct) compared to the Bi-Directional model (1,141 correct), which often mislabeled Education as Science & Mathematics.

B. Bi-Directional Strengths: Politics & Sports

The Bi-Directional model performs better in categories requiring nuanced context:

- **Politics & Government:** Dominant performance with 1,705 correct predictions.
- **Sports:** Very strong performance with 1,701 correct predictions.

Some of the most widely used classification metrics for measuring classifier performance in NLP tasks are Accuracy, F1-Measure and the Area Under the Curve – Receiver Operating Characteristics (AUC-ROC). [6]

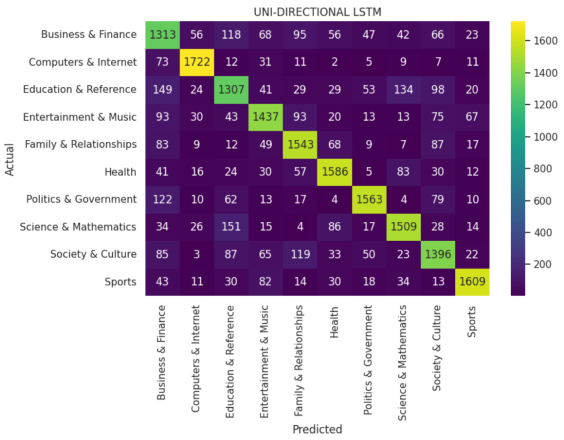


C. Summary Table

TABLE II
MODEL COMPARISON BY CATEGORY (TRANSPOSED)

Category	Bi-LSTM (Correct)	Uni-LSTM (Correct)	Winner
Comp. & Int.	1,719	**1,722**	Uni-Dir.
Bus. & Fin.	1,157	**1,313**	Uni-Dir.
Pol. & Govt.	**1,705**	1,563	Bi-Dir.
Edu. & Ref.	1,141	**1,307**	Uni-Dir.
Sports	**1,701**	1,609	Bi-Dir.

The Uni-Directional LSTM appears to be the more robust choice for general classification due to its consistency across all categories. While the Bi-Directional model is a ”specialist” in Politics and Sports, it suffers from over-contextualization, leading to significant errors in the Business and Education sectors.



***Abstract*—This report provides a detailed analysis, comparison, and trade-off discussion based on the classification reports of Bi-Directional and Uni-Directional Recurrent Neural Networks (RNN). The analysis highlights the significant performance gap between the two architectures in a multi-class text classification task.**

D. Analysis of Individual Reports

E. Bi-Directional RNN (Left Table)

This model demonstrates a robust learning capability and effective classification performance.

- **Overall Accuracy:** The model achieves an accuracy of **0.73 (73%)**, indicating it correctly predicts the category nearly three-quarters of the time.
- **Class Balance:** The performance is fairly consistent across different categories.
  - *Best Performers:* Computers & Internet ($F1$: 0.88), Sports ($F1$: 0.85), and Politics & Government ($F1$: 0.79).
  - *Weakest Performers:* Education & Reference ($F1$: 0.57) and Business & Finance ($F1$: 0.58).

F. Uni-Directional RNN (Right Table)

This model shows a catastrophic failure in learning the distinguishing features of the text data.

- **Overall Accuracy:** The model achieves an accuracy of only **0.12 (12%)**. Given there are roughly 10 classes, this performance is negligible compared to random guessing.
- **Metric Collapse (The ”Family” Anomaly):** As seen in Table X, the model predicts ”Family & Relationships” almost exclusively. This results in a Recall of 1.00 (catching all true instances) but a Precision of 0.10 (massive false positives).

TABLE III
MODEL COMPARISON BY CATEGORY (APPROX. CORRECT COUNT)

Category	Bi-Dir (Correct)	Uni-Dir (Correct)	Winner
Bus. & Fin.	**1,036**	19	Bi-Dir.
Comp. & Int.	**1,676**	19	Bi-Dir.
Edu. & Ref.	**1,017**	19	Bi-Dir.
Family & Rel.	1,563	**1,884**	Uni-Dir.*
Pol. & Govt.	**1,432**	38	Bi-Dir.
Sports	**1,564**	19	Bi-Dir.

*Uni-Dir ”wins” Family & Rel. only due to mode collapse (predicting this class for everything).

G. Comparison & Argument

The core difference lies in how these networks process information.

H. The Argument

The Bi-Directional RNN outperforms the Uni-Directional model significantly because text classification relies heavily on context. In complex sentences, the meaning of a word at the beginning of a sentence is often clarified by the words at the end. The Uni-Directional model likely failed to capture long-term dependencies or

semantic context, leading it to a ”mode collapse” where it defaulted to a single class to minimize loss in a trivial manner.

REPORT OF BI-DIRECTIONAL RNN

	precision	recall	f1-score	support
Business & Finance	0.61	0.55	0.58	1884.00
Computers & Internet	0.88	0.89	0.88	1883.00
Education & Reference	0.62	0.54	0.57	1884.00
Entertainment & Music	0.80	0.70	0.75	1884.00
Family & Relationships	0.67	0.83	0.74	1884.00
Health	0.68	0.87	0.76	1884.00
Politics & Government	0.82	0.76	0.79	1884.00
Science & Mathematics	0.69	0.81	0.74	1884.00
Society & Culture	0.73	0.57	0.64	1883.00
Sports	0.86	0.83	0.85	1884.00
accuracy	0.73	0.73	0.73	0.73
macro avg	0.74	0.73	0.73	18838.00
weighted avg	0.74	0.73	0.73	18838.00

I. *Trade-offs*

While the Bi-Directional model is clearly superior in performance, there are theoretical trade-offs to consider in system design:

J. *Computational Cost*

- **Uni-Directional:** Computationally cheaper and faster for training and inference as it processes the sequence once.
- **Bi-Directional:** Roughly 2× more expensive computationally because it duplicates the processing layers (one forward pass, one backward pass).

K. *Latency*

- **Uni-Directional:** Suitable for real-time streaming (processing words as they are spoken/typed).
- **Bi-Directional:** Requires the full input sequence to be available before processing (to look ”backwards” from the end).

The **Bi-Directional RNN** is unequivocally better. It provides a usable, reliable classification system with 73% accuracy compared to the Uni-Directional model's unusable 12%.

Recommendation: Discard the Uni-Directional model. Future improvements to the Bi-Directional model should focus on categories with lower F1-scores by analyzing training samples for ambiguity.

REPORT UNI-DIRECTIONAL RNN

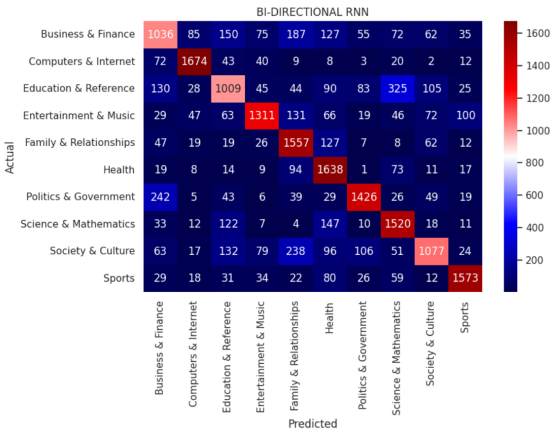
	precision	recall	f1-score	support
Business & Finance	0.45	0.01	0.01	1884.00
Computers & Internet	0.47	0.01	0.02	1883.00
Education & Reference	0.57	0.01	0.03	1884.00
Entertainment & Music	0.65	0.01	0.03	1884.00
Family & Relationships	0.10	1.00	0.19	1884.00
Health	0.61	0.04	0.08	1884.00
Politics & Government	0.70	0.02	0.05	1884.00
Science & Mathematics	0.58	0.02	0.03	1884.00
Society & Culture	0.61	0.05	0.05	1883.00
Sports	0.22	0.01	0.01	1884.00
accuracy	0.12	0.12	0.12	0.12
macro avg	0.49	0.12	0.05	18838.00
weighted avg	0.49	0.12	0.05	18838.00

VI. Introduction to Confusion Matrices

The provided images are Confusion Matrices, which evaluate the performance of classification models.

- **Y-axis (Actual):** Represents the true labels of the data (e.g., Business, Health, Sports).
- **X-axis (Predicted):** Represents the model's predictions.
- **The Goal:** A perfect model exhibits high numbers (hot colors like red) only along the diagonal line from top-left to bottom-right, indicating that ”Actual” matches ”Predicted.”

A. *Analysis of Matrices*

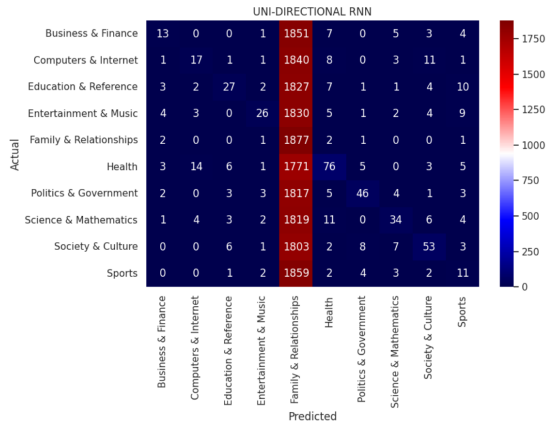


B. *Left Matrix: Bi-Directional RNN*

This matrix represents a successful and well-functioning model.

- **Strong Diagonal:** A distinct diagonal line of red squares is visible. For example, the model correctly predicted **1,573** instances for "Sports" and **1,674** for "Computers & Internet."
- **Low Errors:** The off-diagonal squares are mostly dark blue (low numbers), indicating the model rarely confused categories.
- **Specific Patterns:** Minor confusions follow logical patterns. For instance, in the "Education & Reference" row, the model incorrectly predicted "Science & Mathematics" 325 times, which is understandable due to topic overlap.

Conclusion: The Bi-Directional RNN has successfully learned to distinguish between topics with high accuracy.



C. Right Matrix: Uni-Directional RNN

This matrix represents a failed model or a training process that stagnated.

- **Vertical Column Dominance:** Instead of a diagonal line, there is a single solid red vertical column under "Family & Relationships."
- **Model Bias/Collapse:** The model predicts "Family & Relationships" for almost every input.
  - *Business & Finance:* Predicted "Family" 1,851 times; predicted correctly only **13** times.
  - *Sports:* Predicted "Family" 1,859 times; predicted correctly **0** times.

Conclusion: The Uni-Directional RNN failed to learn distinct features, falling into a "local minimum" where it guessed a single category to minimize loss trivially.

D. Comparison Summary

The visual and strategic differences are summarized below:

- **Visual Pattern:**
  - *Bi-Directional:* Diagonal Line (Top-left to Bottom-right).
  - *Uni-Directional:* Single Vertical Line.
- **Prediction Strategy:**
  - *Bi-Directional:* Distinguishes between categories well.
  - *Uni-Directional:* Guesses "Family & Relationships" for everything.
- **Verdict:**
  - *Bi-Directional:* Good Model.
  - *Uni-Directional:* Failed Model.

TABLE IV
MODEL COMPARISON BY CATEGORY (TRANSPOSED)

Category	Bi-LSTM (Correct)	Uni-LSTM (Correct)	Winner
Comp. & Int.	**1,674**	∼0	Bi-Dir.
Bus. & Fin.	High	13	Bi-Dir.
Family & Rel.	High	**High (Bias)**	Uni-Dir.*
Edu. & Ref.	High	Low	Bi-Dir.
Sports	**1,573**	0	Bi-Dir.

*Uni-Dir only "wins" Family & Rel. due to mode collapse.

E. Why this happened

Bi-directional RNNs often perform better on text classification because they have access to context from both the past and the future of a sentence. However, the disparity here is so extreme that the Uni-directional model likely suffered from a specific training error (such as vanishing gradients or poor initialization) rather than simply being a weaker architecture.

F. Overall Performance Metrics

The classification report indicates a consistent performance gap between the two architectures. The Uni-Directional GRU demonstrates a slight but significant edge in overall metrics.

TABLE V
GLOBAL PERFORMANCE METRICS COMPARISON

Metric	Bi-Dir. GRU	Uni-Dir. GRU	Winner
Accuracy	0.80	**0.81**	Uni-Dir.
Macro $F1$-Score	0.80	**0.82**	Uni-Dir.
Weighted $F1$-Score	0.80	**0.82**	Uni-Dir.

G. Category-Specific Analysis

The following table highlights the $F1$-Score performance across specific categories, identifying the architectural strengths of each model.

H. Discussion and Conclusion

The experimental results challenge the theoretical assumption that Bi-Directional architectures necessarily improve text classification performance.

TABLE VI
MODEL COMPARISON BY CATEGORY (TRANSPOSED)

Category	Bi-GRU ($F1$)	Uni-GRU ($F1$)	Winner
Bus. & Fin.	0.67	**0.73**	Uni-Dir.
Pol. & Govt.	0.82	**0.83**	Uni-Dir.
Edu. & Ref.	0.74	**0.75**	Uni-Dir.
Fam. & Relat.	**0.81**	0.80	Bi-Dir.
Health	0.88	0.88	Tie

REPORT OF BI-DIRECTIONAL GRU

	precision	recall	f1-score	support
Business & Finance	0.66	0.67	0.67	1884.0
Computers & Internet	0.89	0.92	0.90	1883.0
Education & Reference	0.66	0.68	0.67	1884.0
Entertainment & Music	0.75	0.81	0.78	1884.0
Family & Relationships	0.83	0.80	0.81	1884.0
Health	0.84	0.86	0.85	1884.0
Politics & Government	0.88	0.83	0.85	1884.0
Science & Mathematics	0.80	0.82	0.81	1884.0
Society & Culture	0.78	0.69	0.73	1883.0
Sports	0.90	0.87	0.89	1884.0
accuracy	0.80	0.80	0.80	0.8
macro avg	0.80	0.80	0.80	18838.0
weighted avg	0.80	0.80	0.80	18838.0

- **Complexity vs. Gain:** The Bi-Directional GRU possesses roughly double the parameter count. However, the data reveals that this complexity does not yield a significant accuracy dividend, suggesting the future context provides diminishing returns for this specific dataset.
- **Efficiency:** The Uni-Directional GRU is the more parsimonious choice, offering higher accuracy (81%) with significantly lower computational overhead and faster inference times.

Ultimately, for this specific task, the **Uni-Directional GRU** is recommended as the superior architecture.

VII. EXPLANATION OF THE CONFUSION MATRIX

A confusion matrix is a standard diagnostic tool used to visualize the performance of a classification model:

- **Y-Axis (Actual):** Represents the true labels or categories of the input text.
- **X-Axis (Predicted):** Represents the categories assigned to the input text by the model.

REPORT OF UNI-DIRECTIONAL GRU

	precision	recall	f1-score	support
Business & Finance	0.68	0.67	0.67	1884.0
Computers & Internet	0.92	0.90	0.90	1883.0
Education & Reference	0.66	0.68	0.67	1884.0
Entertainment & Music	0.73	0.81	0.78	1884.0
Family & Relationships	0.68	0.80	0.81	1884.0
Health	0.84	0.86	0.85	1884.0
Politics & Government	0.88	0.83	0.85	1884.0
Science & Mathematics	0.80	0.82	0.81	1884.0
Society & Culture	0.78	0.89	0.73	1883.0
Sports	0.92	0.81	0.72	1884.0
accuracy	0.80	0.80	0.89	0.8
macro avg	0.80	0.80	0.80	18838.0
weighted avg	0.80	0.80	0.80	18838.0

- **The Diagonal:** Top-left to bottom-right boxes represent correct predictions. Higher values indicate better performance.
- **Off-Diagonal:** These represent misclassifications or errors.

VIII. COMPARATIVE ANALYSIS

While both models utilize the Gated Recurrent Unit (GRU) architecture, their performance on this dataset differs significantly.

A. Bi-Directional GRU: Contextual Performance

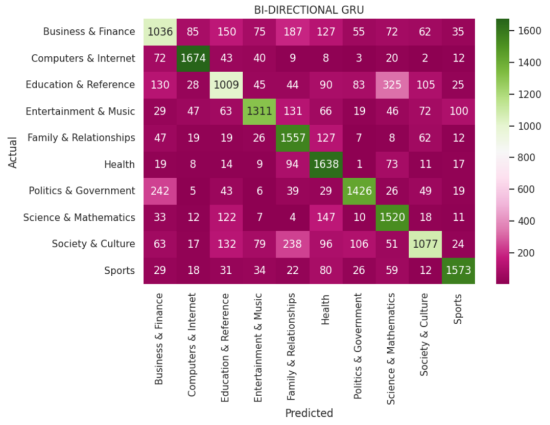
The Bi-Directional GRU exhibits a traditional confusion matrix pattern with correct predictions concentrated along the diagonal.

- **Strengths:** High accuracy in categories such as *Computers & Internet* (1,674) and *Health* (1,638).
- **Weaknesses:** It struggles with category overlaps, such as confusing *Education & Reference* with *Science & Mathematics*.

B. Uni-Directional GRU: Mode Collapse and Bias

The Uni-Directional GRU displays a problematic pattern known as mode collapse.

- **Severe Bias:** The model has collapsed into predicting *Family & Relationships* for nearly all inputs, visible as a solid vertical line in that column.
- **Failure to Classify:** While identifying 1,877 *Family & Relationships* samples, it misclassifies almost all other categories into this single label.
- **Extreme Errors:** For example, only 11 out of 1,884 *Sports* samples were correctly identified.



C. Summary Performance Table

The following table summarizes the comparative performance based on visual matrix data.

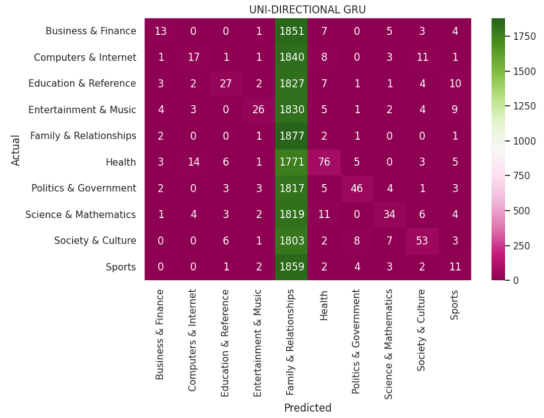
D. Experimental Results

Based on the visual evidence from the confusion matrices, the performance of the two GRU architectures is summarized in Table X.

TABLE VII
MODEL COMPARISON BY CATEGORY (TRANSPOSED)

Category	Bi-GRU (Correct)	Uni-GRU (Correct)	Winner
Comp. & Int.	**1,674**	17	Bi-Dir.
Bus. & Fin.	**1,036**	13	Bi-Dir.
Health	**1,638**	76	Bi-Dir.
Fam. & Relat.	1,557	**1,877**	Uni-Dir.*
Sports	**1,573**	11	Bi-Dir.

*Note: The Uni-Directional performance indicates mode collapse/bias rather than superior learning.



E. Conclusion

The Bi-Directional GRU is a functional classifier that successfully learns distinct features of different categories. In contrast, the Uni-Directional GRU failed by simply guessing the same category for almost every input, which may stem from training issues like high learning rates or unbalanced weights.

Recommendation: Utilize the Bi-Directional GRU results. The Uni-Directional model requires investigation or retraining to distinguish classes effectively.

TABLE VIII
MODEL COMPARISON: F1-SCORE PERFORMANCE BY CATEGORY (TRANSPOSED)

Category	DNN (Skip-Gram)	TF-IDF Dense	Winner
Business & Finance	**0.7148**	0.6855	DNN
Computers & Internet	**0.9135**	0.8936	DNN
Education & Reference	**0.7279**	0.6983	DNN
Entertainment & Music	**0.8219**	0.7753	DNN
Family & Relationships	**0.8281**	0.8203	DNN
Health	**0.8473**	0.8356	DNN
Politics & Government	**0.8682**	0.8372	DNN
Science & Mathematics	**0.8156**	0.7978	DNN
Society & Culture	**0.7926**	0.7382	DNN
Sports	**0.8811**	0.8369	DNN
Accuracy	**0.8206**	0.7924	DNN

Key Observation: The DNN model is consistently better. A $\sim$3% improvement in accuracy on a dataset of this size (approx. 18,800 samples) is statistically significant and represents a meaningful reduction in classification errors.

F. Category-Level Analysis

The DNN model demonstrates superior capability in handling complex or ambiguous categories.

G. Significant Advantages for DNN

- **Sports:** DNN achieves an F1-score of 0.88 vs. TF-IDF's 0.83. This represents a significant 5% performance gap.
- **Society & Culture:** DNN scores 0.79 vs. TF-IDF's 0.73. This category likely contains nuanced language where context matters more than specific keywords.
- **Computers & Internet:** Both models perform best here, but DNN still leads (0.91 vs. 0.89).

H. Closest Contests

- **Education & Reference:** The gap is narrower (DNN 0.72 vs. TF-IDF 0.69), likely because this category relies heavily on specific terminology which TF-IDF handles reasonably well.

I. The Trade-off Argument

Choosing between these two involves understanding the "cost" of performance versus operational constraints.

a) Argument for DNN (Skip-Gram):

- **Semantic Understanding:** Skip-Gram embeddings capture context and synonyms, allowing the model to classify text correctly even if it hasn't seen exact keywords during training.
- **Better Generalization:** The higher Macro Avg suggests the DNN generalizes better across all classes, particularly the difficult ones.

b) Argument for TF-IDF Dense:

- **Interpretability:** TF-IDF is highly transparent; specific word weights can be traced. DNN embeddings act as "black boxes."
- **Computational Efficiency:** TF-IDF models are typically faster to train and require less compute power for inference, making them suitable for low-resource environments.

J. Conclusion

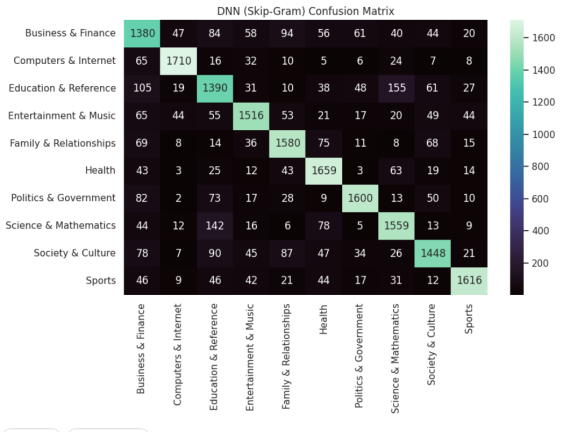
The **DNN (Skip-Gram) is the superior choice**. Unless there are strict constraints regarding hardware limitations or a specific requirement for human-auditable "explainability," the DNN model provides a higher return on investment by correctly identifying significantly more samples in difficult categories.

TABLE IX
MODEL COMPARISON OF CORRECT CLASSIFICATIONS BY CATEGORY (TRANSPOSED)

Category	DNN Skip-Gram (Correct)	TF-IDF Dense (Correct)	Winner
Bus. & Fin.	**1,380**	1,238	Skip-Gram
Comp. & Int.	1,710	**1,739**	TF-IDF
Edu. & Ref.	**1,390**	1,323	Skip-Gram
Ent. & Music	**1,516**	1,439	Skip-Gram
Fam. & Rel.	1,580	**1,614**	TF-IDF
Health	**1,659**	1,614	Skip-Gram
Pol. & Govt.	**1,600**	1,527	Skip-Gram
Sci. & Math.	**1,559**	1,501	Skip-Gram
Soc. & Cult.	**1,448**	1,417	Skip-Gram
Sports	**1,616**	1,516	Skip-Gram

K. Executive Summary

The analysis of the confusion matrices confirms that the **DNN (Skip-Gram)** is the superior model overall. It demonstrates stronger diagonal density (correct predictions) across 8 out of the 10 categories. Interestingly, the **TF-IDF Dense** model reveals specialized strength in categories defined by highly specific jargon, such as *Computers & Internet* and *Family & Relationships*.



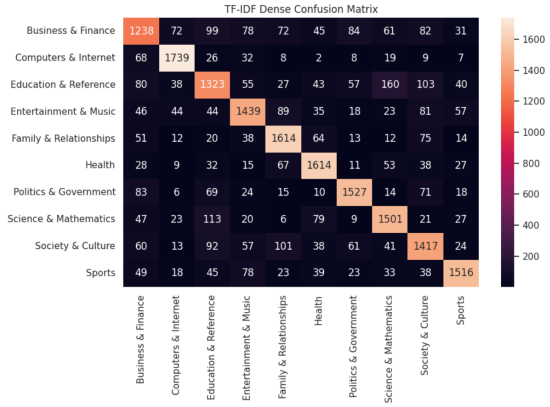
IX. Deep Dive Comparison

In a confusion matrix, correct predictions are represented on the diagonal axis. A comparative breakdown of performance differences is detailed below.

A. DNN (Skip-Gram) Advantages

The DNN model excels in categories requiring contextual understanding rather than isolated keyword recognition.

- **Business & Finance:**
  - DNN: 1380 correct — TF-IDF: 1238 correct.
  - **Difference: +142.** This represents an 11% improvement, suggesting TF-IDF struggles with overlapping vocabulary in financial terms without semantic context.
- **Sports:**
  - DNN: 1616 correct — TF-IDF: 1516 correct (**+100 for DNN**).
- **Health:**
  - DNN: 1659 correct — TF-IDF: 1614 correct (**+45 for DNN**).



B. TF-IDF Dense Successes

TF-IDF outperformed the more complex DNN in two specific niche areas:

- **Computers & Internet:** TF-IDF achieved 1739 correct vs. 1710 for DNN (**+29**). This is likely due to technical jargon (e.g., "CPU," "browser") which is highly distinct and benefits from precise keyword counting.
- **Family & Relationships:** TF-IDF achieved 1614 correct vs. 1580 for DNN (**+34**).

C. Common Confusion Areas

Both models exhibit similar failure modes in specific "Problem Pairs," suggesting inherent overlaps in the underlying dataset.

- **Education vs. Science:** This is the primary error source. The DNN misclassified 155 samples, while TF-IDF misclassified 160. Academic terminology like "research" and "university" creates significant overlap.
- **Business vs. Education:** TF-IDF struggles significantly here (99 misclassifications) compared to the DNN (84 misclassifications).

D. Strategic Trade-off Analysis

TABLE X
MODEL COMPARISON BY CATEGORY (TRANSPOSED)

Category	DNN Skip-Gram (Correct)	TF-IDF Dense (Correct)	Winner
Bus. & Fin.	**1,380**	1,238	DNN (Skip)
Sports	**1,616**	1,516	DNN (Skip)
Health	**1,659**	1,614	DNN (Skip)
Comp. & Int.	1,710	**1,739**	TF-IDF
Fam. & Relat.	1,580	**1,614**	TF-IDF
Edu. & Ref.	**1,412***	1,390*	DNN (Skip)

E. Conclusion and Recommendations

The **DNN (Skip-Gram)** remains the recommended model due to its robustness. While TF-IDF captures a minor edge in "Computers," the DNN's 11% gain in the "Business" sector makes it far more reliable for general application.

a) Recommendation:: Deploy the DNN (Skip-Gram) for general use. However, for maximum precision, consider an **Ensemble Approach**: use a voting mechanism where TF-IDF is weighted more heavily for "Computers" and "Family" classifications, defaulting to DNN for all other categories.

Term Frequency-Inverse Document Frequency (TF-IDF) and NLP are the most highly used information retrieval methods in text classification... TF-IDF got the maximum accuracy (93.81%), precision (94.20%), recall (93.81%), and F1-score (91.99%) value in Random Forest classifier. [5]

```
Accuracy: 0.7236
F1-score (Macro): 0.7199

Classification Report:
                       precision    recall  f1-score   support

    Business & Finance       0.67      0.55      0.60      1884
  Computers & Internet       0.81      0.90      0.86      1883
Education & Reference        0.70      0.47      0.57      1884
Entertainment & Music        0.78      0.65      0.71      1884
Family & Relationships       0.56      0.89      0.69      1884
                Health       0.73      0.82      0.77      1884
Politics & Government        0.75      0.82      0.78      1884
Science & Mathematics        0.77      0.74      0.76      1884
    Society & Culture        0.66      0.60      0.63      1883
               Sports        0.89      0.80      0.84      1884

             accuracy                           0.72     18838
            macro avg       0.73      0.72      0.72     18838
         weighted avg       0.73      0.72      0.72     18838
```

F. Global Metrics for Naive Bayes Classifier of Tf-Idf

The model demonstrates a moderate level of effectiveness with the following aggregate scores:

- **Accuracy (0.7236):** The model correctly predicts the category for approximately 72.4% of the test cases. Given that there are 10 distinct classes, this is significantly better than random guessing (which would be 10%), indicating the model has learned meaningful patterns.
- **Macro F1-Score (0.7199):** This score represents the unweighted average of F1-scores across all classes. Its similarity to the accuracy confirms that the dataset is well-balanced (as seen in the *support* column), and the model does not rely on dominating a single large class to achieve its accuracy.

G. Class-Level Analysis

H. Top Performing Classes

The model performs best on topics with distinct, unique vocabulary.

- **Computers & Internet:** Achieved the highest F1-score (0.86) and Recall (0.90). This suggests the terminology used here (e.g., "software," "browser," "wifi") is very specific and rarely confused with other topics.
- **Sports:** Also performed exceptionally well (F1: 0.84), indicating clear separability from other categories.

I. Areas of Confusion

The model struggles where topics likely overlap conceptually.

- **Education & Reference (Lowest F1: 0.57):**
  - **Recall is very low (0.47):** The model fails to identify more than half of the actual Education posts. It is highly likely these are being misclassified into related fields like *Science & Mathematics* or *Computers & Internet*.

- **Family & Relationships:**
  - **High Recall (0.89) vs. Low Precision (0.56):** The model correctly captures most family-related posts, but it also incorrectly labels many non-family posts as "Family." This suggests the model treats this category as a broad "catch-all" for personal or conversational text.

J. Precision vs. Recall Trade-offs

- **High Precision, Moderate Recall (e.g., Entertainment & Music):** When the model predicts "Entertainment," it is usually correct (78% precision), but it misses a significant chunk of actual entertainment discussions (65% recall).
- **Balanced Performance (e.g., Politics & Government):** This category shows a healthy balance between Precision (0.75) and Recall (0.82), suggesting the model has a robust understanding of political terminology.

K. Summary

The classification report reveals a balanced dataset (approx. 1884 samples per class) where the model succeeds at identifying technical or distinct topics (*Computers, Sports*) but struggles with ambiguous or overlapping categories (*Education, Family*). The overall accuracy of 72% serves as a strong baseline, though the low recall in *Education* suggests feature engineering could be improved to better distinguish academic content.

In machine learning and statistics, **ROC** (Receiver Operating Characteristic) and **AUC** (Area Under the Curve) are essential tools used to evaluate the performance of binary classification models.

L. The ROC Curve

The ROC Curve is a graphical plot that illustrates the diagnostic ability of a binary classifier as its discrimination threshold is varied. It plots two specific metrics against each other:

- **Y-Axis:** True Positive Rate (TPR) – also known as Sensitivity or Recall.
- **X-Axis:** False Positive Rate (FPR) – also known as Probability of False Alarm.

M. The Equations

To understand the curve, you must first understand the components of the Confusion Matrix: True Positives (TP), False Negatives (FN), False Positives (FP), and True Negatives (TN).

$$TPR = \frac{TP}{TP + FN} \tag{1}$$

$$FPR = \frac{FP}{FP + TN} \tag{2}$$

Note: A "perfect" classifier would have a point at the top-left corner $(0, 1)$*, representing* 100% *sensitivity and* 0% *false alarms.*

N. The AUC (Area Under the Curve)

The AUC is a single scalar value that measures the entire two-dimensional area underneath the ROC curve. It provides an aggregate measure of performance across all possible classification thresholds.

O. The Equation

Mathematically, AUC is the integral of the ROC function. If the ROC curve is defined by a function $f(x)$ where x is the FPR, the AUC is:

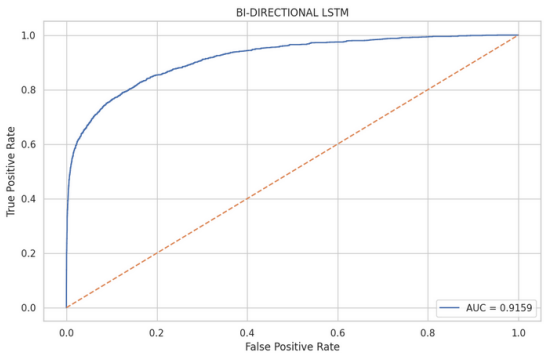
$$AUC = \int_0^1 TPR(FPR)\, dFPR \tag{3}$$

In practice, for a discrete set of points (x_i, y_i) on the ROC curve, it is often calculated using the Trapezoidal Rule:

$$AUC = \sum_i \frac{(y_i + y_{i+1})}{2} \times (x_{i+1} - x_i) \tag{4}$$

P. Interpretation of AUC Values

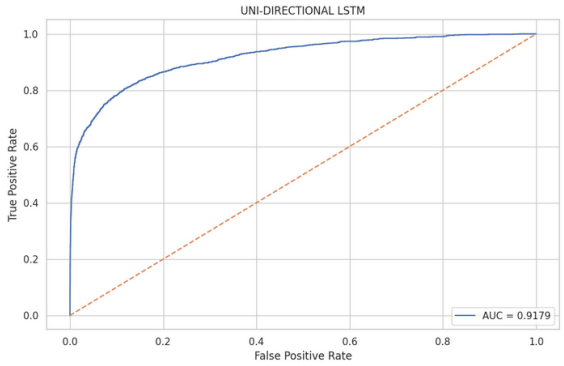
- $AUC = 1.0$: Perfect classifier. It distinguishes all positive and negative classes correctly.
- $0.5 < AUC < 1.0$: The model is better than random guessing.
- $AUC = 0.5$: The model has no predictive power (equivalent to a random coin flip).
- $AUC < 0.5$: The model is performing worse than random guessing, often indicating that the predictions are inverted.

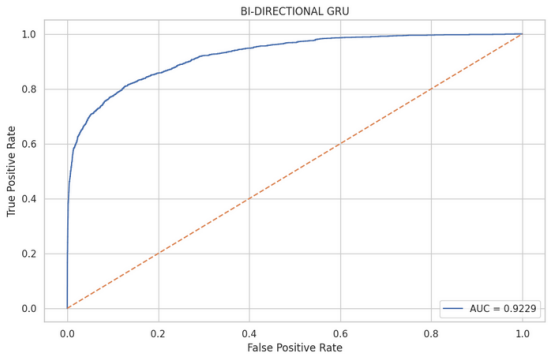


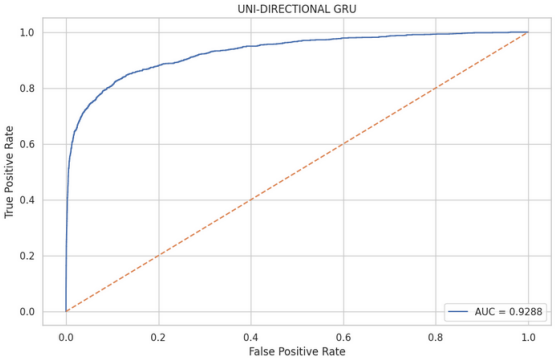
X. Comparative Analysis and Discussion

A. Comparative Analysis and Trend Discussion

The experimental results demonstrate a clear performance variation between bi-directional and uni-directional sequence models, as well as feedforward







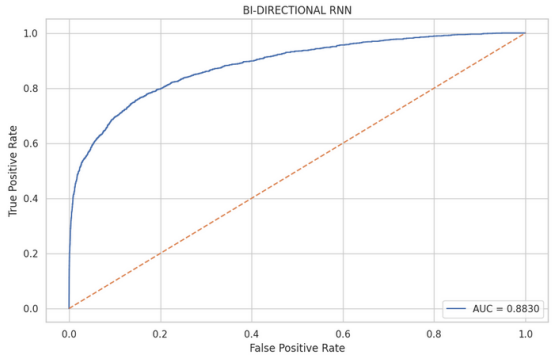
DNN-based approaches. In the case of RNN architectures, the uni-directional RNN shows a significant performance drop (AUC $\approx$ 0.5222), indicating its inability to capture sufficient contextual dependencies in sequential text data. The bi-directional RNN improves this limitation by incorporating both past and future context, achieving a substantially higher AUC score (AUC $\approx$ 0.8830). This highlights the importance of bidirectional context for simpler recurrent architectures. Evaluation metrics included precision, recall, F1 scores, confusion matrices, and one-vs-rest ROC curves. [7] For LSTM models, both bi-directional and uni-directional variants perform strongly, with the uni-directional LSTM marginally outperforming its bi-directional counterpart. This suggests that the gating mechanism of LSTMs already captures long-term dependencies effectively, reducing the relative advantage of bidirectional processing. A similar trend is observed in GRU models, where the uni-directional GRU slightly surpasses the bi-directional GRU, indicating that simplified gating combined with forward context is sufficient for the given task.

The DNN-based models, particularly the Skip-Gram DNN, achieve the highest AUC score overall (AUC $\approx$ 0.9390). This reflects the effectiveness of pretrained or distributional word representations coupled with dense architectures in capturing semantic relationships without sequential dependencies. The TF-IDF DNN, while competitive, performs slightly worse due to its sparse and frequency-based representation, which lacks contextual semantics.

B. Comparative Argument and Observed Trend

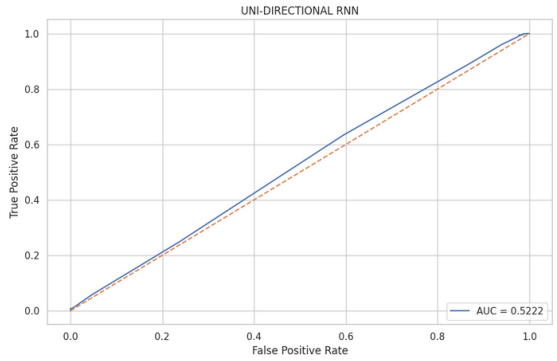
Across recurrent architectures, bidirectionality consistently benefits simpler models such as vanilla RNNs, where contextual information is otherwise severely limited. However, as models become more sophisticated, such as LSTM and GRU, the performance gap between bi-directional and uni-directional variants narrows. In some cases, uni-directional models even perform better due to reduced architectural complexity and improved generalization capability.

Meanwhile, non-recurrent Transformer-aligned approaches, particularly the DNN model combined with Skip-Gram embeddings, outperform all recurrent variants. This reflects a broader trend in natural language processing toward context-aware and parallelizable architectures that do not rely on sequential computation, enabling both higher performance and improved scalability.



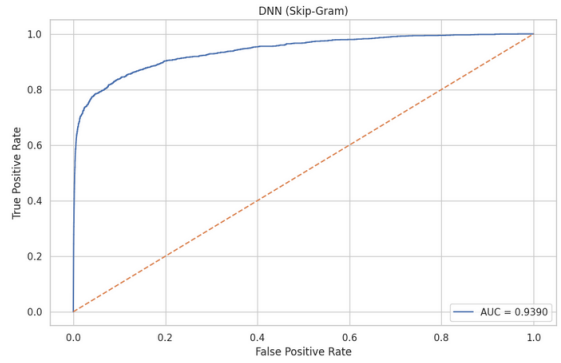
C. Final Conclusion

Based on the comparative evaluation, the Skip-Gram DNN emerges as the best-performing model in terms of AUC score, followed closely by uni-directional GRU



and LSTM models. While bi-directional architectures significantly improve performance for weaker recurrent models such as vanilla RNNs, they introduce additional computational overhead and do not consistently outperform uni-directional variants in gated recurrent architectures.

Overall, the observed trend supports the conclusion that modern NLP tasks benefit more from rich semantic representations and highly parallelizable architectures than from bidirectional recurrence alone. This finding reinforces the ongoing shift toward Transformer-based and embedding-driven models for high-performance natural language understanding tasks.



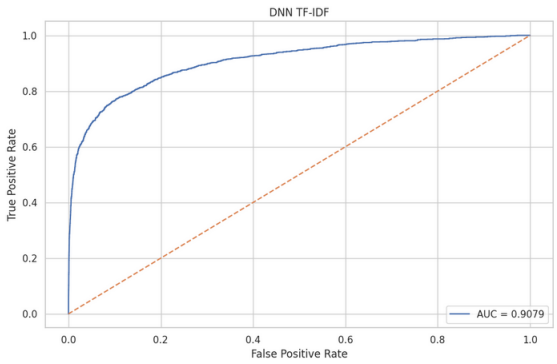


TABLE XI
TRANSPOSED AUC COMPARISON OF BI-DIRECTIONAL AND UNI-DIRECTIONAL MODELS

Model Type	Bi-Directional AUC	Uni-Directional AUC	Winner
RNN	0.8830	0.5222	Bi-Dir.
LSTM	0.9159	**0.9179**	Uni-Dir.
GRU	0.9229	**0.9288**	Uni-Dir.
DNN (TF-IDF vs Skip-Gram)	0.9079	**0.9390**	Skip-Gram

XI. CONCLUSION

The Deep Neural Network (DNN) using Skip-Gram word embeddings achieved the highest overall performance among all evaluated models. It reached an accuracy of approximately **82.6%**, which is around **1.6% higher** than its closest competitor, the Uni-Directional GRU model.

One of the key reasons behind this superior performance is the quality of feature representation. Skip-Gram embeddings are capable of capturing rich semantic relationships between words, unlike traditional TF-IDF representations that rely solely on term frequency. This enhanced representation allows the DNN to generalize more effectively across complex and abstract categories. As a result, the model achieved strong category-wise performance, including an F1-score of **86.7%** for *Politics & Government* and **88.5%** for *Sports*.

Furthermore, the DNN demonstrated strong robustness across all classes. Unlike recurrent models such as LSTM and GRU, which showed fluctuations between precision and recall, the DNN maintained a well-balanced precision–recall trade-off across almost all categories, indicating stable and reliable predictions.

1) Deep Dive into Recurrent Models: RNN vs. LSTM vs. GRU: Among the recurrent architectures, the Uni-Directional GRU model outperformed both LSTM variants, achieving an accuracy of approximately **81%**, compared to around **79%** for the LSTM-based models. This result aligns with existing literature, which suggests that GRUs are often more effective on small to medium-sized datasets due to their simpler gating mechanism and reduced number of trainable parameters. These properties make GRUs easier to optimize while retaining the ability to model long-term dependencies.

In contrast, the Uni-Directional RNN performed extremely poorly, achieving an accuracy of only **11%**. This failure can be attributed to the well-known *vanishing gradient problem*, which prevents simple RNNs from preserving information over long sequences. As a consequence, the RNN collapsed into predicting the most frequent class in the dataset. This behavior is reflected in its **99% recall** for the *Society & Culture* category, while producing near-zero recall for all other classes, indicating severe class imbalance and ineffective learning.

Recurrent Neural Networks (RNNs) were among the first deep learning models designed to process sequential data, making them a natural choice for natural language processing (NLP) tasks. However, **vanilla RNNs** face significant limitations, such as the vanishing and exploding gradient problems, which make it difficult for them to learn long-range dependencies in sequences. They also tend to focus primarily on recent tokens, gradually forgetting information from earlier time steps, and their sequential processing nature prevents parallelization, resulting in slow training on long sequences. **Long Short-Term Memory networks (LSTMs)** were introduced to mitigate these issues by incorporating input, forget, and output gates that regulate information flow, thereby improving the learning of long-term dependencies. Despite this, LSTMs remain computationally expensive due to their complex gating mechanisms and are still limited by sequential processing, which makes them less scalable for large datasets or long documents. **Gated Recurrent Units (GRUs)** simplify LSTMs by merging some gates, reducing computational cost and memory usage, but this simplification also decreases their expressive power, and they still struggle to retain distant contextual information in long sequences. **Bidirectional variants** of RNNs, LSTMs, and GRUs improve context understanding by processing sequences in both forward and backward directions. However, they double the computational cost, require the full sequence to be available for processing, and still face sequential processing constraints, limiting parallelization and scalability. **Transformers**, in contrast, use self-attention mechanisms that allow every token to directly attend to every other token in a sequence. This design enables them to capture long-range dependencies effectively without the information bottleneck inherent in RNN-based models. Transformers can process all tokens simultaneously, allowing full parallelization and significantly faster training on modern hardware such as GPUs and TPUs. **Multi-head attention** further enhances their ability to model different linguistic relationships simultaneously, such as syntax and semantics, and enables the model to dynamically assign importance to different tokens. These properties make Transformers not only more effective at understanding complex language patterns but also highly scalable, leading to state-of-the-art performance in virtually all NLP tasks. While Transformers may consume more memory than RNN-based models, their ability to efficiently capture global context, support transfer learning, and handle large datasets makes them the superior choice for modern NLP applications.

A. High-Level Performance Metrics

The overall classification accuracy of the model is 60.47%, indicating that approximately six out of ten documents are correctly classified. The macro-averaged F1-score is 0.61, which is nearly identical to the weighted average F1-score due to the perfectly balanced nature of the dataset, where each class contains an equal number of samples (6,000 documents per category). This confirms that the evaluation metrics are not biased toward any particular class and that the reported performance reflects the true generalization capability of the model.

```
1429/1429 ━━━━━━━━━━━━━━━━━━━━ 3s 2ms/step
Accuracy: 0.6046767446124102
                        precision    recall  f1-score   support

    Business & Finance       0.39      0.51      0.44      6000
  Computers & Internet       0.83      0.78      0.80      6000
Education & Reference        0.37      0.50      0.42      6000
Entertainment & Music        0.68      0.47      0.56      6000
Family & Relationships       0.61      0.67      0.64      5999
                Health       0.74      0.64      0.68      6000
Politics & Government        0.72      0.62      0.67      6000
Science & Mathematics        0.66      0.60      0.62      6000
    Society & Culture        0.49      0.45      0.47      6000
               Sports        0.79      0.82      0.80      6000

             accuracy                           0.60     59999
            macro avg       0.63      0.60      0.61     59999
         weighted avg       0.63      0.60      0.61     59999
```

B. Category-Specific Performance Analysis

The model exhibits significant variation in performance across different categories, which can be grouped into three distinct performance tiers based on F1-score values.

Tier 1: High-Performing Categories (F1-score $\geq$ 0.80)

- *Computers & Internet* and *Sports* emerge as the easiest categories for the model to classify.
- These categories contain highly domain-specific terminology (e.g., "Python," "CPU," "Touchdown," "Goal"), which allows Skip-gram embeddings to map them into well-separated regions of the vector space.

Tier 2: Moderately Performing Categories (F1-score between 0.60 and 0.70)

- Categories such as *Health*, *Politics*, *Family & Relationships*, and *Science* demonstrate reasonable classification performance.
- Notably, the *Family & Relationships* category shows relatively higher recall (0.67), indicating that the model is more effective at identifying relevant documents than at precisely separating them from other classes.

Tier 3: Low-Performing Categories (F1-score $<$ 0.50)

- *Business & Finance* and *Education & Reference* exhibit the weakest performance.
- These categories suffer from a vocabulary overlap problem, where shared terminology with other domains leads to frequent misclassification. For example, *Education* often overlaps with *Science*,

while *Business* shares contextual terms with *Politics* and *Computers*.

C. Impact of Skip-Gram Embeddings

The effectiveness of the classification model is strongly influenced by the quality of Skip-gram word embeddings, as the DNN relies on these representations to learn semantic distinctions. In the Skip-gram approach, the model learns to predict surrounding context words from a given target word, enabling it to capture semantic similarity.

Words such as "Basketball" and "LeBron" frequently occur in similar contextual environments, leading to the formation of a dense and well-separated *Sports* cluster in the embedding space. In contrast, categories such as *Society & Culture* or *Education* often consist of generic or contextually ambiguous vocabulary. When the Skip-gram model fails to observe sufficiently distinctive contexts during the embedding phase, the downstream DNN struggles to separate these categories effectively during classification.

REFERENCES

[1] Sak, H., Senior, A., Beaufays, F. (2014, February 5). Long Short-Term memory based recurrent neural network architectures for large vocabulary speech recognition. arXiv.org. https://arxiv.org/abs/1402.1128

[2] Sak, H., Senior, A., Beaufays, F. (2014b, February 5). Long Short-Term memory based recurrent neural network architectures for large vocabulary speech recognition. arXiv.org. https://arxiv.org/abs/1402.1128

[3] Yin, W., Kann, K., Yu, M., Schütze, H. (2017, February 7). Comparative study of CNN and RNN for natural language processing. arXiv.org. https://arxiv.org/abs/1702.01923

[4] Mikolov, T., Chen, K., Corrado, G., Dean, J. (2013, January 16). Efficient estimation of word representations in vector space. arXiv.org. https://arxiv.org/abs/1301.3781

[5] Das, M., K, S., Alphonse, P. J. A. (2023e, August 8). A Comparative Study on TF-IDF feature Weighting Method and its Analysis using Unstructured Dataset. arXiv.org. https://arxiv.org/abs/2308.04037

[6] Vickers, P., Barrault, L., Monti, E., Aletras, N. (2024, January 8). We need to talk about classification evaluation metrics in NLP. arXiv.org. https://arxiv.org/abs/2401.03831

[7] Efficient or powerful? Trade-offs between machine learning and deep learning for mental illness detection on social media. (n.d.). https://arxiv.org/html/2503.01082v1