CHAPTER 1

INDTRODUCTION

This chapter includes the following subtopics, namely: (1) Rationale; (2) Theoretical Framework; (3) Conceptual Framework/Paradigm; (4) Statement of the problem; (5) Hypothesis; (6) Assumption; (7) Scope and Delimitation; and (8) Importance of the study.

1.1 Rationale

Fashion trend forecasting has become important in the fashion industry due to its ability to help fashion companies in predicting the emerging trend and remain competitive. Forecasting accuracy becomes crucial in order to forecast keeping up with trends and meet the market demands. Despite its importance, fashion trend forecasting has significant issues, particularly due to difficulties in capturing trends that change rapidly over time.

Existing studies on fashion trend forecasting mainly utilized RNN methods, such as LSTM in encoder-decoder architectures [1], [7], [13], [14]. The notable model, like KERN, achieving promising result on FIT data set [1]. However, there are several areas that could be improved in this model. This includes further optimizing its performance by exploring various method configurations and tuning hyperparameters, as well as enhancing its ability to handle longer and more complex dynamic patterns of fashion trends.

Based on these challenges, this study intends to address them with several enhancements. First, this study will experiment with alternative methods, such as GRU [15], for both the encoder-decoder framework, as well as combine LSTM and GRU. Second, this study will explore attention mechanisms to be integrated into the model, which has been proven effective in improving the model's ability to handle dependencies between trend points [7], [16]. Lastly, this study will conduct hyperparameter tuning [17] to enhance the models' performance beyond current parameter configurations. By implementing these enhancements, this study aspires to contribute to the ongoing development of more accurate and adaptable fashion forecasting models.

1.2 Theoretical Framework

This study integrates several theoretical foundations to enhance the accuracy of the KERN model in fashion trend forecasting. The main theoretical frameworks supporting this research are as follows:

1. Fashion Trend Forecasting
   Fashion trend forecasting involves forecasting future styles and consumer preferences

[18]. As trends are time-dependent and influenced by global events, economic conditions, and cultural activities [19], fashion companies must anticipate consumer demands [20]. Trends can refer to styles, fabrics, patterns, and colors [21].

Traditionally, trend forecasting relied on designers' artistic vision and market observations. And yet, these methods are costly and subjective. The rise of the internet and social media has enabled data-driven approaches, providing extensive user-generated fashion records. However, forecasting faces two key challenges: (1) selecting and processing relevant data and (2) choosing an appropriate forecasting method [22].

Sales data, commonly used for forecasting, is often influenced by pricing, customer reviews [22], and company-specific factors, making it less publicly accessible. In contrast, social media offers a vast, publicly available dataset with continuous fashion insights. Several datasets, such as the FIT dataset, provide structured fashion-related information with fine-grained elements spanning multiple years [1].

2. Time-Series Data in Forecasting

   Time series data presents observations, such as characteristics and behavior, sequentially over time. This data also captures trends, patterns, and temporal dependencies. Analyzing these patterns and trends could provide useful information for the decision-making process [14]. Time-series usually contain three components: 1) trend to represent the general movements of values over time, without considering seasonality or irregularities, 2) seasonality to indicating repeated pattern that occur at regular time intervals, 3) residuals, which indicate the noise in the sequence and could disturb the forecasting process and seasonality pattern if their value is big enough [23].

   Time-series data is widely used in forecasting tasks, including fashion trend forecasting, as both share a common objective: to forecast the future pattern based on historical data. Fashion trends exhibit time-dependent patterns, where certain fashion items or fashion attributes could gain or lose their popularity over time. On the other hand, the quality of data is affecting the way the model learns and predicts the trend movements [23]. In fashion forecasting, obtaining high-quality data is particularly difficult due to several reasons. These include reliance on sales records that are not always publicly available, inconsistent fashion element labeling, or limited coverage of fashion attributes such as item type, fabric, color, and pattern. Given these issues, selecting an appropriate dataset and forecasting method is crucial for improving the accuracy of trend forecasting.

3. Encoder-Decoder Architecture

   Encoder-Decoder Architecture is one of the popular approaches to forecasting time-series data. It has a main purpose: to compress important information from source data (in the form of a sequence) into a fixed-length vector [24]. This architecture has several advantages. First, it maps historical time-series inputs to future values to enable effective long-term forecasting. Second, it incorporates both the time-series input and associated sequence information into a unified model [1]. These characteristics support the architecture's ability to handle more diverse forecasting scenarios.

Most studies in fashion trend forecasting, such as [1], [7], and [13], use LSTM as the method in encoder-decoder architecture. It is due to the ability of LSTM to memorize important information and forget unimportant information through its gating mechanism. LSTMs perform better than traditional RNNs by solving the vanishing gradient problem, which happens with long input sequences [25].

Another variant of RNN is known as GRU. It uses fewer gates compared to LSTM while also being able to control information flow [26]. With fewer gates, GRU computes faster than LSTM, making it a promising alternative method for encoder-decoder architectures. This study explores utilizing GRUs in the encoder-decoder architecture while also investigating the combination of both LSTM and GRU within the same framework.

4. KERN model

   One implementation of the encoder-decoder architecture is the KERN model [1], where it processes fashion trend data from social media. The model introduces knowledge features, which have been shown to improve forecasting performance. There are two types of knowledge features: internal and external knowledge. Internal knowledge captures the similarity relations between fashion element sequences, while the external knowledge incorporates affiliation relations between fashion elements based on their taxonomy. However, the KERN model exclusively uses LSTM as the forecasting method, and some of the knowledge features may not consistently support forecasting tasks on certain future time periods.

5. Attention Mechanism

   The attention mechanism is an extension of the encoder-decoder model that seeks the position of the most relevant information in the source data . This mechanism allows the model to focus on using relevant information to generate the next target word, values, etc., in the decoder part [24]. However, this approach may lead to inefficient computation and inaccurate information when dealing with long time steps of historical data in a time-series forecasting task [16]. There are two emerging attention mechanisms that could address this issue: multimodal [16] and sliding window attention [7]. Both multimodal and sliding window attention attend to historical data within fixed-length periods. The former processes entire periods, while the latter moves step by step across fixed window lengths. Both mechanisms extract information and fuse it into knowledge, which the decoder utilizes for forecasting the target output.

A previous study has demonstrated the efficiency of sliding window attention in fashion trend forecasting [7]. On the other hand, multimodal attention with periodical historical data has been useful in sales and electricity forecasting [16]. This study aims to systematically evaluate both mechanisms under the same conditions to assess their impact on fashion trend prediction.

6. Hyperparameter Tuning
   Hyperparameter tuning enhance performance of a machine learning model (ML) by selecting the optimal value of the models' hyperparameters [17]. For example, the learning rate can affect the model's ability to reach the minimum value [17], while optimizer impacts the model's ability to minimize the loss function during training [27]. This study conducted the hyperparameter tuning using sequential approach, where each parameter values are selected and tested iteratively to identify each parameter impacts on model performance. The selected parameter values are informed in the experimental settings.

1.3 Conceptual Framework/Paradigm

This study aims to enhance the KERN model's forecasting ability by introducing GRU in the architecture, exploring GRU-LSTM combinations in the architecture, integrating an attention mechanism, and optimizing key hyperparameters. Figure 1.1 shows the conceptual framework of this study, including several variables and processes.

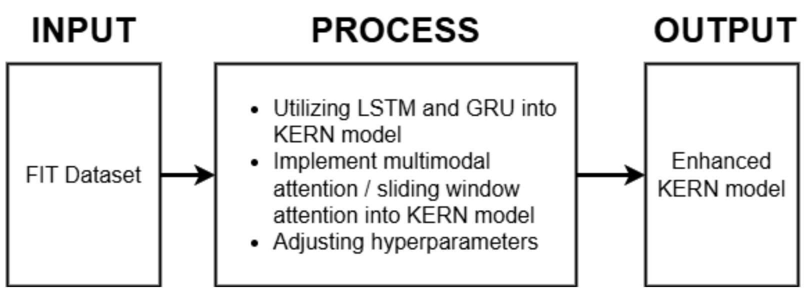

Figure 1.1: This study's conceptual framework

1. Input
   Input in this study consists of sequences of fashion elements distributed into multiple user groups. The sequence values are integer, with time steps ranging from 0 to 23 to indicate trends over a one-year period and total trends of 5 years. Fashion elements and user groups are written in string format and will be embedded before being processed as input into the KERN model.

2. Process

   In general, the encoder processes input sequences, along with embedded fashion elements and user groups. The encoder generates a final hidden state and then passes it on to the decoder. The decoder uses this information to generate predicted trend sequences for future time steps.

   This study introduces multiple enhancements to the KERN model through a structured process. First, the model is modified to incorporate both LSTM and GRU in different encoder-decoder pairings. Next, an attention mechanism is applied between the encoder and decoder to create an enhanced hidden state for the decoder by utilizing both the encoder's output and the final hidden state. Finally, hyperparameter tuning is performed to optimize the performance of the enhanced KERN model.

3. Output

   Output in this study is an enhanced KERN model capable of forecasting fashion trends over various future time horizons, such as 6-month or 12-month. The forecasted trends are represented as sequences in order to maintain the same structured format from the input data. These predicted sequences capture the expected fashion element popularity over time, allowing more accurate and adaptive trend forecasting.

1.4 Statement of the Problem

This study addresses the challenge of improving the accuracy of the KERN model in fashion trend forecasting. The original KERN model has limitations because it relies on a single method for its encoder-decoder architecture and the use of fixed hyperparameter values. This study aims to answer several main questions, which are as follows:

1. How does incorporating different encoder-decoder pairings affect the KERN model's performance in predicting fashion trends?

2. How does utilizing the attention mechanism impact the KERN model's performance in trend prediction?

3. How does choosing parameters affect the performance of the enhanced KERN model?

1.5 Objective and Hypotheses

1.5.1 Objective

Three objectives are created from the questions in the previous section. They are in the following order:

1. To enhance the KERN model for fashion trend forecasting by integrating different encoder-decoder methods

2. To address long-term dependency information loss in the KERN model by adding the attention mechanism

3. To improve the KERN model's learning efficiency and forecasting accuracy through hyperparameter tuning

1.5.2 Hypotheses

The proposed enhancement, which include using various methods on the encoder-decoder architecture, incorporating attention mechanisms, and exploring different hyperparameter configurations, are expected to significantly improve the accuracy and performance of the KERN model.

1.6 Assumption

The proposed enhancements were implemented on the following assumptions:

1. Implementing GRU on the KERN model is expected to yield better forecasting performance due to using a smaller number of gates than the LSTM.

2. The KERN model performance can be improved by incorporating multimodal attention to capture a better information from historical data by attending it in equal length periods.

3. The KERN model performance could be improved more better by using sliding window attention instead of multimodal attention. It was due to the attention process applied on overlapped periods of historical data that may provide richer contextual information compared to non-overlapping sequences.

4. The intertwined hyperparameters, including the learning rate and optimizer, can be further optimized to improve the model performance.

5. The exclusion of the knowledge feature in the KERN model may lead to improved performance compared to its inclusion.

1.7 Scope and Delimitation

This study focuses on enhancing the existing forecasting model, KERN, by introducing several improvements. This study also utilizes the same dataset as for the original KERN model and represents fashion trends from 2014 to 2019. Only three parameters are selected for hyperparameter tuning, while other potential parameters are left unchanged and are not explored in this study.

1.8 Significance of the Study

The main contribution of this study is to introduce enhancements with the aim of improving the performance of the KERN model. The first enhancement involves adding GRU alongside the LSTM method and combining both methods within an encoder-decoder architecture. The second enhancement introduces an attention mechanism to the model architecture. There are two kinds of attention mechanisms explored. The first is multimodal attention, which applies attention to equally length-segmented sub-sequences. The second is sliding window attention, which applies attention to dynamically generated subsequences based on the sliding window technique. The last enhancement focuses on hyperparameter tuning to optimize the model's performance.