**CIND860 Advanced Data Analytics Project:**

**Fashion MNIST - Evaluating the Efficacy of CNN against Traditional ML Models**

**Literature Review, Data Description and Approach**

**By: Hasib Chowdhury (500439135)**

**Supervisor: Dr. Tamer Abdou**

**GitHub Link: https://github.com/Hasib147/CIND860-Capstone-Project**

**Date of Submission: October 23, 2023**

**Revised Abstract: (modification from the original abstract)**

For this project, I will use the Fashion MNIST dataset (<https://www.kaggle.com/datasets/zalando-research/fashionmnist>) from the Kaggle website to conduct the research to fulfill the “Advanced Data Analytics Project (CIND860)” requirements.

There are a total of 70,000 images for this dataset, which is split 60,000 for training and 10,000 for testing (comprising of 28x28 grayscale images). This exact dataset is also built-in in the keras library, which shows the exact images (shirts, pants, sneakers, etc.) in its pixel grayscale form as this is not available on Kaggle. The Kaggle dataset only has 2 .csv files for training and testing for the 784 pixels in the dataset (ranging from 1 to 255 in darkness of the image).

The theme that has been chosen for this project is the deep learning theme (specifically image classification on various Fashion attires) and the technique being used is Convolutional Neural Network (CNN) on the Fashion MNIST dataset.

Some of the research questions this project will go into detail is what models of the CNN is the most efficient to use. For example, Is CNN's performance practical when compared to traditional machine learning techniques such as Random Forest, SVM or XGBoost? As well as How the CNN's performance compared to other models like LeNet-5, VGGNet, ZFNet, or ResNet? *“Fashion businesses in general have used CNN on their e-commerce platforms to solve many problems such as clothes recognition, clothes search and recommendation. A core step for*

*all of these implementations is image classification. However, clothes classification is a challenge task as clothes have many properties, and the depth of clothes categorization is highly*

*complicated.” [1].*

One other thing I will investigate is if we are able to identify the significant features that can accurately predict the classification of the fashion attire dataset? And how different fashion attires such as shirts, pants, sneakers, etc. have an effect on the entire dataset. Also the performance of different evaluation measures, such as Accuracy, Recall, and Precision vary? (within different layers of the dataset)

This project will look into various properties for the clothes such as which sizes are used (small, medium, large, XL) in certain attires as well as how different attires suit different types of people. This project will investigate is if there is a discrepancy in the CNN model when it comes to different age and genders like different fashion attire such as T-shirs/tops for Men and Women compared to kids or teenagers*.* Lastly, this project will see how the CNN compares with the SVM as mentioned in ***other*** research papers and other types of models when it comes to accuracy and precision of the dataset, as well as data labelling of different types of attire.

I will use Python as the main programming language. I will also look at which specific models of the CNN architecture are the most commonly used when evaluating the Fashion MNIST dataset in deep learning models. Throughout the fashion industry and also in Fashion e-commerce and in online retail such as Amazon and E-bay, the market has been growing in recent years and the “CNN model in particular has been shown greater efficiency in image c1assification” [3]. This is what this project will look into as the main technique that is going to be used throughout the project.

**Introduction:**

“The fashion market has changed dramatically over the last 30 years, resulting in an evolution in that industry. Understanding customer tastes and better-directing sales are the way to increase profit” [3]. “The rise of internet business lets people buy their clothes through websites, faster and easier. The introduction of methods to improve user’s experience when searching for items in these platforms is decisive” [3]. In platforms such as Amazon and E-bay, many consumers are buying their clothing and footwear online in recent years compared to the years prior where they were buying in-person at the store. The trend from retail clothes shopping in store to online shopping has drastically changed, mostly for the better good but there can be issues with online shopping such as fraud with credit cards, but most transactions are secure.

I have also used ChatGPT for this part of the project, when prompted “In 3-4 paragraphs describe Fashion MNIST specifically for Fashion Business industry if possible. And maybe include Amazon or E-bay as platforms for online shopping based on a literature review”. The ChatGPT-generated text indicated that Fashion MNIST dataset holds significant importance for the fashion business industry, serving as a cornerstone in the field of computer vision and machine learning. In addition, “Fashion MNIST is essentially a collection of grayscale images representing various fashion items, such as clothing, footwear, and accessories that we use in our daily lives. Each image is associated with a corresponding label, classifying the item it represents. This dataset plays a pivotal role in training and testing machine learning models, making it a valuable resource for fashion businesses seeking to leverage cutting-edge technologies.” [4]

In terms of how Amazon and E-bay play a role with Fashion MNIST in the business market. The AI states that “With the growth of e-commerce platforms like Amazon and eBay, the utilization of Fashion MNIST has gained particular relevance. These platforms, among others, employ machine learning and computer vision techniques to enhance the shopping experience for customers. By using Fashion MNIST, they can develop recommendation systems that suggest products based on a user's previous purchases and preferences, leading to increased customer engagement and sales. Additionally, they can implement image recognition algorithms to enable users to search for items by simply uploading a photo, making the shopping process more convenient and intuitive.” [4]

And lastly “In recent years, the fashion industry has witnessed a transformation in the way it operates, with the integration of technology and data-driven approaches. Fashion MNIST serves as a foundational tool for fashion businesses, allowing them to develop and refine machine learning models for image classification, object detection, and even style analysis. By observing the different types of capabilities this dataset can perform, companies can enhance their understanding of consumer trends, streamline inventory management, and ultimately drive growth in the highly competitive online marketplace” [4]. In other words, it plays a key role in the convergence of the fashion and technology market industry and it allows businesses of various sectors (including the clothing industry) to be more up to date when it comes to the innovation in the digital era (especially in online shopping such as Amazon, E-baby, Facebook marketplace, etc.)

**Questions from Module 2:**

* What do you already know about the topic?

I know the Fashion MNIST dataset is quite a popular dataset among deep learning applications, it is mainly used for image classification tasks, and it can be used to train deep learning models, such as convolutional neural networks (CNNs), for tasks related to fashion item recognition such as different clothing for different types of people. I have used this in a previous course (CIND850) in an assignment and would like to learn more about the deep learning applications associated with it.

* What do you have to say critically about what is already known?

What I have to say critically is that I will evaluate the existing body of knowledge that other researchers or practitioners have already done when the dataset first came out. I will also try to figure out the strengths, weaknesses, gaps, and limitations of prior research and what new outcomes can be achieved from doing this research project.

* Has anyone else ever done anything exactly the same?

I am replicating an existing research paper done by someone else involving CNN with the same research questions and outputs but I am using certain techniques that they haven’t used and I am comparing 4 different CNN models in terms of accuracy. I may also try to use precision and recall, which they didn’t do in their research and see if there is an improvement in the results. I will also try to get the same results as outlined in their research paper in terms of accuracy, however in terms of sampling I may use a smaller sample size then they did because of the limitation in the GPU in the Google Collab environment (otherwise the system crashes due to the very large dataset).

* Has anyone else done anything that is related?

Yes, there is several projects based on this dataset on Kaggle and it’s used in a wide variety of deep learning applications, however I will tailor my project mainly towards the CNN aspect of it. Also the theoretical research with Fashion MNIST has been done many times with different people worldwide but they may have not covered certain topics that I will cover, mainly with CNN models. In addition, there is several research papers online available for Fashion MNIST and I have used 3 main sources.

* Where does your work fit in with what has gone before?

My work fits in with what has gone before because I am applying a new method to evaluate this particular CNN model and how it compares to the different types of traditional machine learning algorithms on Fashion MNIST.

* Why is your research worth doing in the light of what has already been done?

There are lots of ongoing and existing research on this particular dataset that is on Kaggle and I will do my best to build on the existing research and the body of knowledge which is already known at this point, that is why I am doing this to shed light on things not mentioned before. As mentioned earlier, one research paper (article) is being replicated but different techniques are being used to see if there is a difference in the results and how other CNN models have any effects on it.

**Literature Review:**

The specific research paper that I will be replicating is the “Classifying Garments from Fashion-MNIST Dataset Through CNNs” taken from the *Advances in Science, Technology and Engineering Systems Journal* article Volume 6, published and made online in February 2021. This research paper is 6 pages long and consists of various CNN models used to interpret Fashion MNIST and its applications on deep learning.

“Convolutional Neural Network models have been shown efficiency in image c1assification. This paper presents four different Convolutional Neural Networks models that used Fashion-MNIST dataset. Fashion-MNIST is a dataset made to help researchers finding models to classify this kind of product such as clothes, and the paper that describes it presents a comparison between the main classification methods to find the one that better label this kind of data.[3]

In other words, this paper addresses the growing online fashion market's need for algorithms capable of identifying garments. Such algorithms can help companies in the clothing sales sector understand customer preferences, tailor marketing campaigns, and enhance the customer experience while shopping online or at the store. Convolutional Neural Networks (CNNs) are known for their efficiency in image classification, and this paper presents four different CNN models applied to the Fashion-MNIST dataset, I plan to use the same 4 CNN models and see how they are similar or different from one another when I try to calculate their efficiency as presented in this paper. The original research evaluated various machine learning models and achieved 89.7% accuracy using SVM. I will try to use the same technique and see if I end up with the same results (or at least close to it). Also in this paper, the authors propose the use of CNNs to label the Fashion-MNIST dataset, aiming to enhance accuracy. The results show that their new CNN model called "cnn-dropout-3" achieves an accuracy of 99.1%, which was the highest out of all the 15 different models that were tested however it maybe the more time consuming process, which I will investigate.

To summarize the introduction, this paper is an extension of work originally presented at the Iberian Conference on Information Systems and Technologies [3]. It emphasizes the changing dynamics in the fashion industry driven by internet business and the importance of understanding customer preferences and improving the user experience. Classifying clothing is part of the broader task of classifying scenes, and automating this process can assist deep learning researchers and provide insights into users' tastes, culture, and financial status. The original work used various AI models and achieved the best result using SVM with 89.7% accuracy. This paper proposes the use of CNNs for labeling the Fashion-MNIST dataset to improve classification accuracy. I will replicate the same method that they used but if I run into any further problems with the coding, I may switch to using precision or recall instead of accuracy if the results are off by a lot, I want to keep the results consistent as it’s the exact same dataset with the same number of labels as well as same training/test sets.

To summarize the background, this paper discusses the concepts of Machine Learning, Feature Learning, and Deep Learning. Feature Learning is essential for building models capable of pattern recognition, and Deep Learning methods, including Convolutional Neural Networks (CNNs), have shown promising results. CNNs are particularly effective for image classification tasks, and this paper highlights the key components of CNNs, such as convolutional layers, pooling layers, and dropout as a technique to mitigate overfitting. I will do this same technique like what they have done such as cnn-droput-1, cnn-dropout-2, etc. each with its own attributes and parameters such as number of epochs, batch sizes, optimizer used, etc. and see if the accuracy is consistent as to what they have given in the summary table of their paper.

In terms of related work, this specific paper has used a grand total of 28 different reference from various sources although each reference has its own aspects that were discussed but all deal with the Fashion MNIST and how it is used in deep learning and in the fashion industry. The paper references previous research in the field of clothing classification and recognition. It mentions works that used context-sensitive grammars, multi-class learners based on Random Forest, and Bidirectional Convolutional Neural Networks for clothing landmark localization and classification. I will try out Random Forest and bidirectional neural network and see how the results are if they differ a lot from the accuracy that they have provided.

In the dataset section of the paper, the Fashion-MNIST dataset is introduced as a drop-in alternative to the original MNIST dataset, containing grayscale images of fashion products. It has the same structure as MNIST but with fashion items instead of digits (valued from 0-9 in testing and training). The dataset is described as having two CSV files, one for training images and one for testing images, each with 785 columns, including a label column. The dataset's organization and structure are discussed for data access. Also the values in this csv are values from 0-255 depending on how light or dark the brightness is on that specific pixel (for example you can have very light top at 0 or very dark top at 255). This paper aims to explore and implement CNN models for clothing classification using the Fashion-MNIST dataset and compares the results with the original research, achieving a notable accuracy improvement. It is a valuable contribution to the field of machine learning and fashion recognition.

And in terms of CNN model usage, this paper presents four different Convolutional Neural Network (CNN) models (as discussed before) developed using Python with Keras and TensorFlow to label the Fashion-MNIST dataset. Training was conducted in a Jupyter notebook with GPU support, and Weights and Biases were used to monitor training and hardware usage. I will be using the same technique as this with Python but instead of Jupyter notebook, I will be using Google Collab instead and it will be done using GPU just like how they did it. The version of Tensorflow that I will be using is 2.13.0, this may differ from what they have used as an older version was used when this paper first got published online in February 2021, also I won’t be using weight/biases when doing my code for this project. Also the dataset will be pulled from Keras library directly just like how it’s done on this paper, I have used “from keras.datasets import fashion\_mnist” as the main command to get the Fashion MNIST, this is by default set at 60,000 samples for training and 10,000 samples for testing.

In terms of the results of the 4 different CNN models (that I will be replicating), from the paper the 4 different models had the following results:

1. **cnn-dropout-1 and cnn-dropout-3:**
   * These models employ two consecutive blocks consisting of convolution, max pooling, and dropout layers. Each block is connected to two fully connected layers, which, in turn, connect to an output layer with ten neurons, each representing a category.
   * The difference between the two models is that cnn-dropout-3 features considerably lower dropout values compared to cnn-dropout-1.
   * The topology of these models includes 44,426 trainable parameters.
2. **cnn-dropout-2:**
   * This model is similar to cnn-dropout-1 but with two convolution layers before each max pooling operation.
   * It contains around 32,340 trainable parameters.
3. **cnn-simple:**
   * cnn-simple is a simpler model with fewer layers, featuring only two convolution layers followed by a fully connected layer, along with dropout and max pooling layers.
   * It has 110,968 trainable parameters.
   * Due to its structure, the image reaches the dense layer with a size of 14x14 pixels (four times the size of other models), resulting in slower training in the dense layer.

I will try with my own 4 models of the same kind (CNN-simple is one type and CNN-dropout-1/2/3 are the 3 other types). All of these models are implemented using Keras Sequential models and use the Rectified Linear Unit (ReLU) activation functions for convolutional and dense layers. Softmax activation is used for the output layer. The optimizer chosen is Adadelta, with a batch size of 128, and the models are trained for 12 epochs. Additionally, image pixel luminosity values are normalized to float numbers between 0 and 1 to enhance results. These models aim to efficiently label the Fashion-MNIST dataset, making them suitable for real-time applications such as online stores and search websites. I am going to use the same sequential model as they did and the same activation function (ReLu for the dense layer and Softmax for ouput). I will also use the same epoch and batch sizes for all 4 models, but the optimizer I may use RMSProp or Adam instead of Adadelta, in terms of seeing if it make a difference but may use what they have done.

**Exploratory Data Analysis:**

When taking into account the variance for the top 10 pixels for the test Fashion-MNIST datastet (10,000 rows and 785 columns). The following are the top 10 results of the pixels according to its variance value (using the Feature Selection Filter method) and leaving the 1st column (label) as the dependent variable:

pixel41 10843.149748

pixel44 10788.700657

pixel42 10584.011999

pixel43 10504.884222

pixel740 10370.564248

pixel741 10355.745779

pixel45 10343.841578

pixel745 10264.832946

pixel746 10047.646965

pixel40 10046.087725

These 10 pixels above represent the most relevant pixels due to the high variance value, the other 784 were below 10,000 and were not considered but still play a key role in the dataset, there were some pixels that had variance values less than 1000 and some very small values closer to zero.

As for the pandas profiling report, it is found in the GitHub account along with the .ipynb file of the code I have so far. This specific report was generated based on the “top\_10\_pixels.csv” file and not the original test dataset due to the very large size and time constraint of generating a report of this magnitude. This report contains details about each specific pixel variable and how it correlates to other variables and the dependent variable as well as the general overview of the report.

**Data Description:**

In this dataset, there are a total of 60,000 images of the training dataset and 10,000 images of the test dataset. The 2 .csv files provided on Kaggle provide the details of all 784 pixels (28x28 in terms of height and width) and each pixel on certain labels range from 0-255 (with 0 being light and 255 being very dark). In terms of the labels of the fashion attire in this dataset, they are ranked according to the numbers they represent:

|  |  |
| --- | --- |
| **Label Number** | **Type of Fashion attire** |
| 0 | T-shirt/Top |
| 1 | Trouser |
| 2 | Pullover |
| 3 | Dress |
| 4 | Coat |
| 5 | Sandal |
| 6 | Shirt |
| 7 | Sneaker |
| 8 | Bag |
| 9 | Ankle Boot |

The above table correlates with the labels that were given on Kaggle. In addition, for the training dataset, there are 6,000 labels each for each attire for a grand total of 60,000 fashion images. For the testing however, its a smaller sample size and only 1,000 labels for each attire for a grand total of 10,000 fashion images. The .csv files do not specify the size of the certain clothing like whether certain shirts or trousers are small, medium, or large or whether any type of footwear such as sandals, sneakers, or ankle boots are size 1, 2, etc. and if its gender specific (for males or females).

**Approach:**

The approach that I made when I extracted this dataset was to first download the 2 .csv files onto my hard drive and use the Fashion MNIST from Kaggle website. I then explored the data in more detail by identifying the different types of variables in it (784 total columns, each representing a specific grayscale pixel). There is no target variables and no data cleaning involved in this process as all the values in the spreadsheet are between 0-255 (from very dark to very bright). There are 6,000 values for each label for a grand total of 60,000 values for training and 1,000 values for each label for a grand total of 10,000 values for testing. The actual dataset for specific grayscale fashion images is imported from the keras library, in google collab it comes from “from keras.datasets import fashion\_mnist” command.

In terms of the steps for approach, the following steps were used:

* **Step 1: Downloading the dataset from Kaggle**
* In this step, the raw data was taken for Fashion MNIST and imported it onto my hard drive.
* **Step 2: Data Processing/Extraction**
* In this step, different types of attributes were observed in the 2 .csv files (no changes needed to be made as the first column was the labels variable ranging from 0-9 and the rest were pixels 1-784 sorted ascending order). There were no missing ‘NA’ values that needed to be cleaned.
* The testing dataset had 1,000 labels for each value (0-9) for 10,000 total rows and training data had 6,000 labels for each value (0-9) for 60,000 total rows. The number of columns stayed the same in both .csv files.
* **Step 3: Test/Train Model**
* By default, it was set to 60,000 for training and 10,000 for testing. This is the best split as it is legitimate. That is what came in Kaggle as well as in the Keras library.
* The specific images for each type of clothing can only be imported in the keras library.
* **Step 4: Develop abstract**
* A revised version of the abstract was re-written with additional research questions that were not discussed in the original.
* **Step 5: Literature review**
* This section is where the literature review was written and was used to explain which source that is being replicated and what specific areas within the article that needs to be replicated the same way or differently.
* The tools and techniques that is going to be used and how is it similar or different from the research article will be discussed
* **Step 6: Initial Results**
* Python is used as the main programming language to determine how the various different types of CNN models will compare with one another.
* Due to the system crashing cause of the very large dataset size, small sample sizes were used as a method to evaluate the initial results. This will only be done for testing, not training however and it was done so by using the variance values of each pixel.
* **Step 7: Developing & Analyzing CNN model**
* GPU will be used to analyze the dataset and see how hidden or outer layers will be a factor when comparing the CNN dropout models with other models, as discussed in source [3].
* In addition, random forest and SVM will be used to see if there are effects on it compared to the regular model and other types of CNN [3].
* **Step 8: Result and Discussion**
* Different types of results within the replicated article such as which machine learning algorithm worked the best will be analyzed and for which situation can be done to make improvements to it or perhaps have the same result will be discussed.
* I also mentioned I am going to use the same 4 CNN model techniques as the article and compare them to see how they are similar to one another and what needs revising (like adding layers or dropouts to the CNN model or even using different batch size or epochs, etc.) [3].
* **Step 9: Recommendation & Conclusion**
* This is the final part of the approach and it’s based on the evidence from the data and the results obtained through the various machine learning models and the 4 different CNN models that the paper compares as well as other factors.

References

[1] Xiao, H., Rasul, K., & Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*.

[2] Kayed, M., Anter, A., & Mohamed, H. (2020, February). Classification of garments from fashion MNIST dataset using CNN LeNet-5 architecture. In *2020 international conference on innovative trends in communication and computer engineering (ITCE)* (pp. 238-243). IEEE.

[3] LEITHARDT, V. (2021). Classifying garments from fashion-MNIST dataset through CNNs. *Advances in Science, Technology and Engineering Systems Journal*, *6*(1), 989-994.

[4] OpenAI. (2023). ChatGPT (GPT-3.5 model) [Large language model]. <https://chat.openai.com/chat>