Sentiment Analysis on Machine Learning Based on HP Taccola Printer Customer Sentiment Data

Spencer Chang
Mountain View High School, Vancouver, Washington, USA

Advisor: Jeffery Sheadel, Hewlett Packard, jeffery.sheadel@hp.com

Abstract

The Trillium printer coming out for the next generation of HP printers comes from the design of the Taccola printer, which had notably bad reviews. In order to make a better printer, HP wanted to capitalize on the weak points of the Taccola printer, keep the strong points, and design Trillium with the best of both worlds. 17000 customer comments from the last 3 months were analyzed to find general sentiment on the product. Using Python machine learning, the comments were run though sentiment models in order to deliver a minimum of 80% of sentiments correct. Using Sentence-Bert, SentenceTransformer, and spatial distance, a 90% accuracy rate was achieved. In conclusion, the main weakness of the Taccola needs to change its app in order to function smoother.

Keywords

Sentiment Analysis, Machine Learning, Python, SentenceTransformer, SentenceBERT

Introduction

From the last 3 months of the Taccola data, comments were analyzed from a database of around 17000 comments on what the customers thought about the Taccola printer. From there, most of the comments were on the printer setup process, which many found to be unclear and tedious, and that if it was not able to connect the first time, then the printer would not respond at all and could not be used. Another prominent issue was the Instant Ink feature from HP. People found the feature to be non-beneficial to themselves, and thought that it was too expensive for what it covered. The third greatest comment problem was the print speed. For about 70% of the comments on print speed, they thought that the printer printed very slowly, and that it was very hard for the printer to begin printing. However, the other 30% stated that the print speed was very fast with no issues at all. This leads one to think that the issue with print speed may be in the connection between the printer and the device printing from.

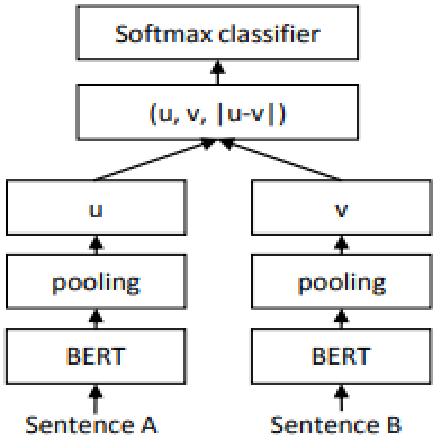


Figure 1: SBERT architecture with classification objecting function. The two BERT networks have tied weights (siamese network structure)

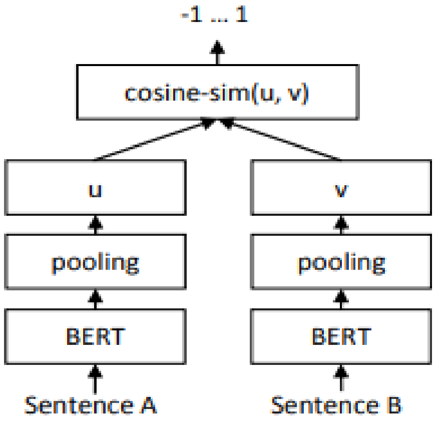


Figure 2: SBERT architecture at inference, for example, to compute similarity scores.

Methodology

The first method used was to take the sentiment of each data using machine learning, and have the data run through the other comments using that. First in order to clean up the data, word labels were changed to number labels - through excel, then separated documents into rows with

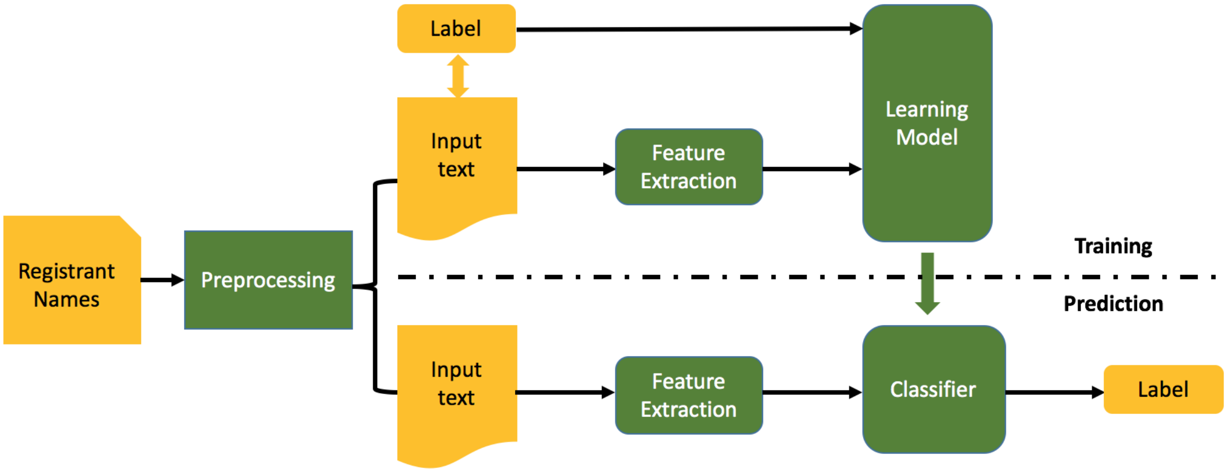


Figure 3: Training Pattern for Machine Learning

labels and those without. The next step would be to delete useless columns. Once the data was cleaned up, the program would be able to look for rows with outliers. For context, data from the Palermo printer was provided, whose comments had 800 comments with the ‘note’ section filled in, which would be the training data. However, this strategy proved to be ineffective as 800 out of 20000 comments did not provide the accuracy that was needed for the training to work.

This led to the second method, which was to use comparison of notes to train the machine learning instead of sentiment. In doing so, the SentenceTransformers model, which is a Python framework, can compute text embeddings. The embedding assigns each word in the cell as a number, then uses cosine similarity to compare how similar phrases are by the distance they are away from the encoded comment, which was used as a base for how the comments should be judged. This method can evaluate the data faster and more exact, using Sentence-BERT(SBERT), which uses “siamese and triplet network structures to derive semantically meaningful sentence embeddings,” meaning that fixed sized vectors could be found, which allows for cosine similarity. Through the cosine method, we are able to find the general sentiment of the dataset, instead of having to rely on the sentiment of all the comments one by one. For training the dataset, we can add more encoders or change which data cell corresponds to what kind of ‘note’.

Data Analysis Methodology

Most of the comments were on the printer setup process, which many found to be unclear and tedious, and that if it was not able to connect the first time, then the printer would not respond at all and could not be used. Another prominent issue was the Instant Ink feature from HP. People found the feature to be non-beneficial to themselves, and thought that it was too expensive for what it covered. The third greatest comment problem was the print speed. For about 70% of the comments on print speed, they thought that the printer printed very slowly, and that it was very hard for the printer to begin printing. However, the other 30% stated that the print speed was very fast with no issues at all. This leads one to think that the issue with print speed may be in the connection between the printer and the device printing from. The fourth largest section of the printer was the customer troubleshooting, which,

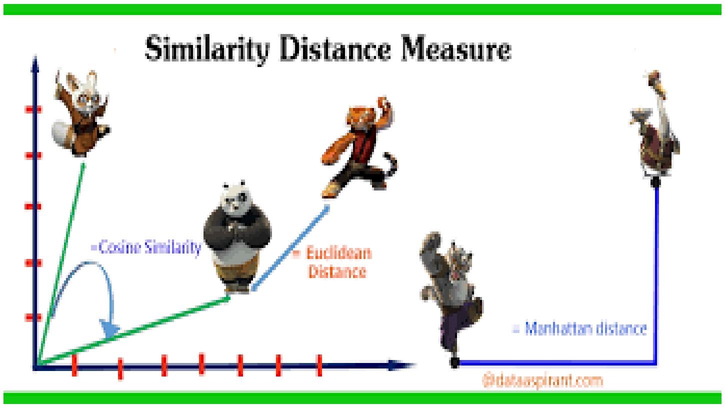


Figure 4: Distance similarity example

generalized, stated that talking to people from customer service did not help, took many hours, and returned no results.

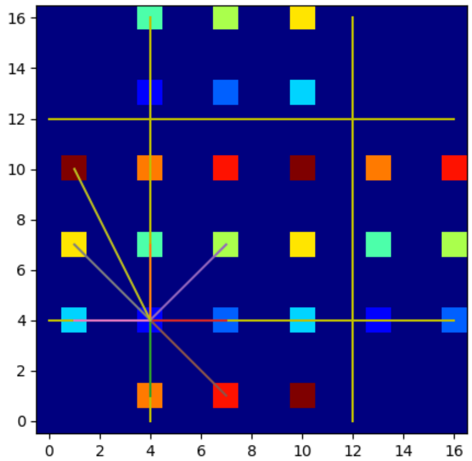


Figure 5a: Matrice for scipy.spatial.distance

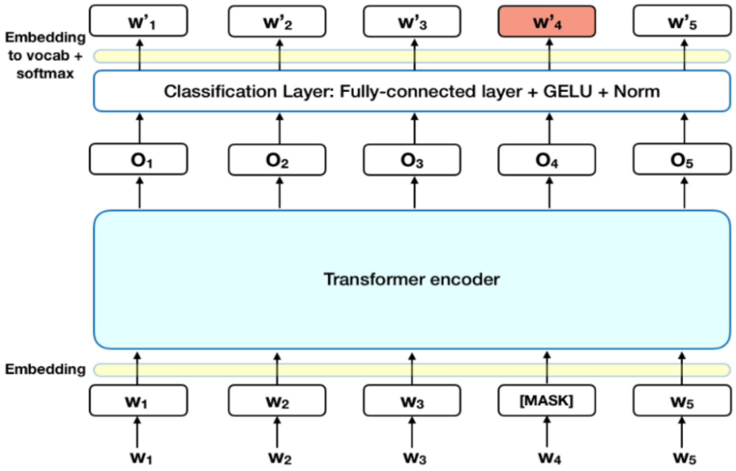
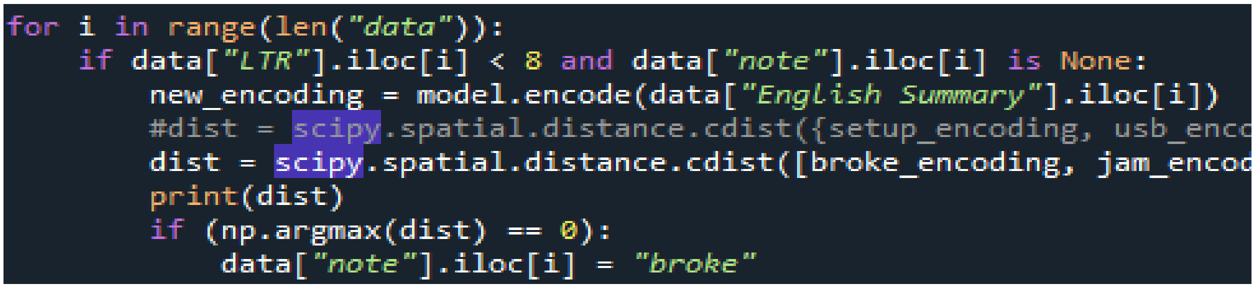

w'1
w'2
w'3
w'4
w'5
Embedding to vocab + softmax
Classification Layer: Fully-connected layer + GELU + Norm
O1
O2
O3
O4
O5
Transformer encoder
Embedding
w1
w2
w3
[MASK]
w5
w1
w2
w3
w4
w5

Figure 5b: encoder for classification layer

Through the imports of scipy.spatial.distance, we are able to find the distance matrix computation from a collection of raw observation vectors stored in a rectangular array. It contains both predicates for checking the validity of distance matrices, both condensed and redundant. Also contained in this module are functions for computing the number of observations in a distance matrix. By using the formula d = √[(x2 – x1)2 + (y2 – y1)2], we are able to solve for the distance between the 'words,' which use the points as the numbers corresponding to each sentence. From there, through the use of scipy.dist, we are able to determine which sentences are closest to the ones we have marked as the encoding.

BERT Example:

15% of the words in each sequence are replaced with a [MASK] token. The model then attempts to predict the original value of the masked words, based on the context provided by the other, non-masked, words in the sequence. In technical terms, the prediction of the output words



```
for i in range(len("data")):
    if data["LTR"].iloc[i] < 8 and data["note"].iloc[i] is None:
        new_encoding = model.encode(data["English Summary"].iloc[i])
        #dist = scipy.spatial.distance.cdist({setup_encoding, usb_enco
        dist = scipy.spatial.distance.cdist([broke_encoding, jam_encod
        print(dist)
        if (np.argmax(dist) == 0):
            data["note"].iloc[i] = "broke"
```

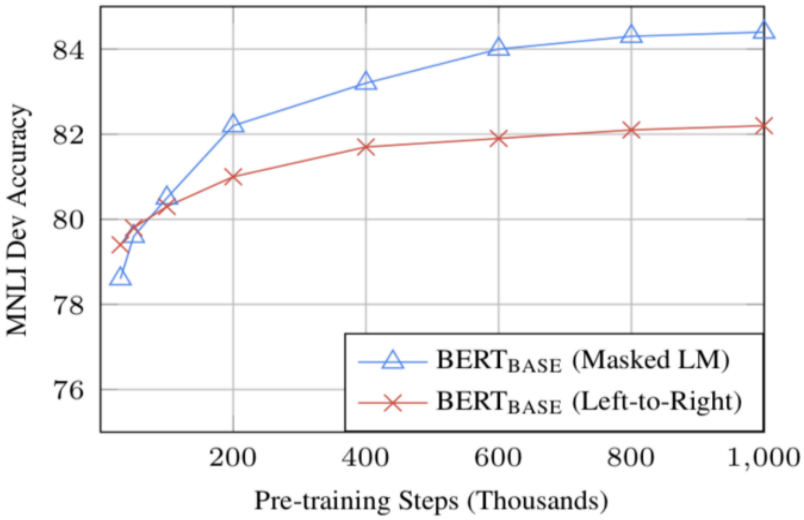
Figure 6: Code fragment of classification

requires adding a classification layer on top of the encoder output. Multiplying the output vectors by the embedding

matrix, transforming them into the vocabulary dimension. Calculating the probability of each word in the vocabulary with softmax. The output of the [CLS] token is transformed into a 2×1 shaped vector, using a simple classification layer (learned matrices of weights and biases).” (Reimers 2019)

The next problem was about how to take the sentiment of each data using machine learning, and have the data run through the other comments using that. First, in order to clean up the data, changing word labels into number labels - through excel, then separated documents into rows with labels and those without. The next step would be to delete useless columns. Once the data was cleaned up, the program would be able to look for rows with outliers. For context, Palermo printer data was provided, whose comments had 800 comments with the ‘note’ section filled in, which would be the training data. However, this strategy proved to be ineffective as 800 out of 20000 comments did not provide the accuracy that was needed for the training to work.

This led to the use of comparison of notes to train machine learning instead of sentiment. In doing so, the SentenceTransformers model was used, which is a Python framework, which can compute text embeddings. The embedding assigns each word in the cell as a number, then uses cosine similarity to compare how similar phrases are by the distance they are away from the encoded comment, which was used as a base for how the comments should be judged. This method is capable of evaluating the data faster and more exact, using Sentence-BERT(SBERT), which uses “siamese and triplet network structures to derive semantically meaningful sentence embeddings,” meaning that fixed sized vectors could be found, which allows for cosine similarity. Through the cosine method, we are able to find the general sentiment of the dataset, instead of having to rely on the sentiment of all the comments one by one.



For training the dataset, we can add more encoders or change which data cell corresponds to what kind of ‘note’. The accuracy of this method is higher and can reach the goal of 90%, compared to the scikit learn model, which could not reach the 90^ threshold.

Figure 7: SentenceBERT Training Steps

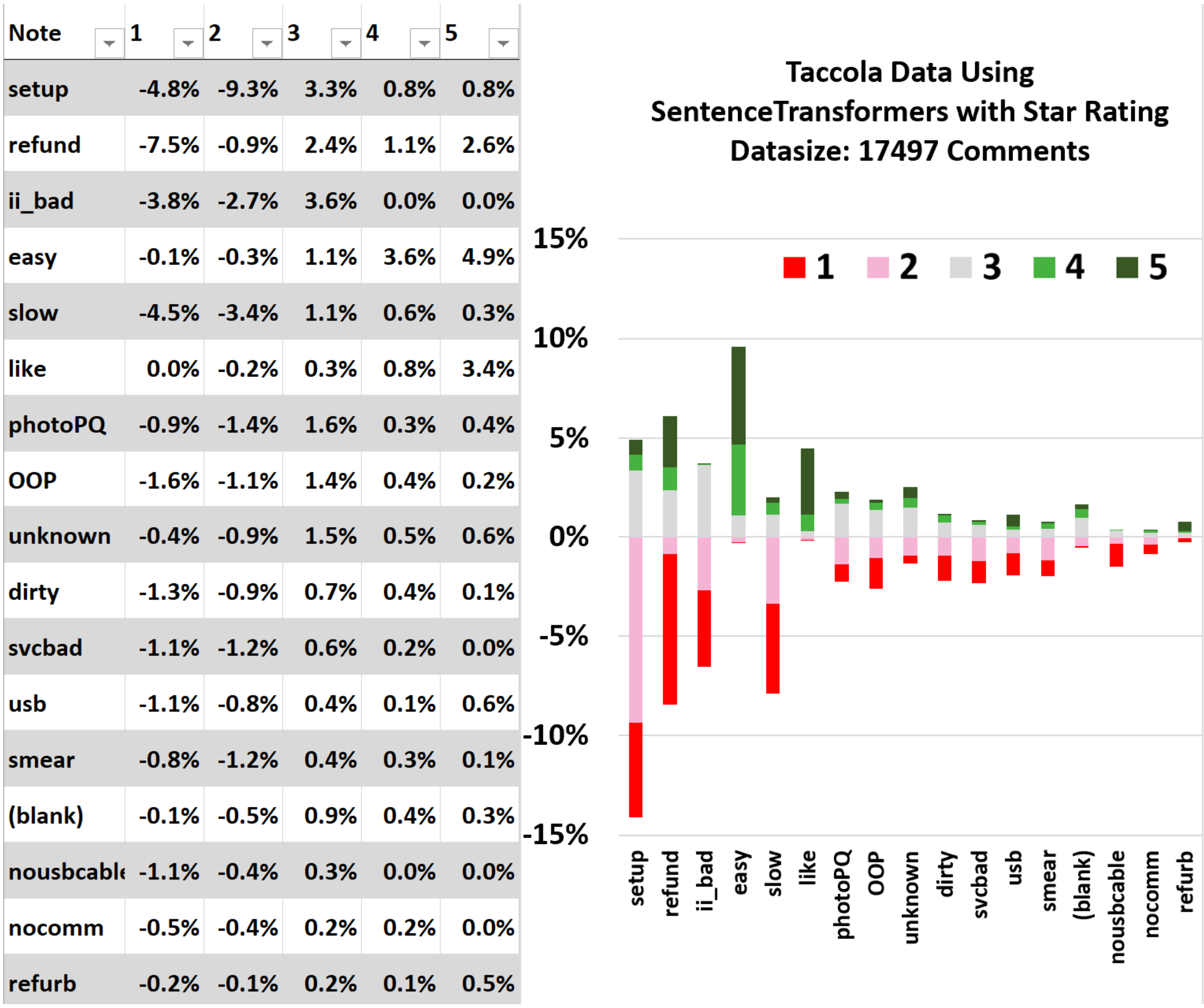
Taccola Data Analysis

On the data below, we can see that compared to the Palermo printer, the Taccola has a higher rate from negative comments photoPQ. With a total of 4.6% of the reviews in photoPQ, there is a mixed statement between good and bad. While the majority of the reviews are at 2 stars for photoPQ, it is important to note that there are also a significant number of 4 and 5 star ratings.

Another outlying problem is the high amount of people complaining about the OOP, out of paper. From the baseline of the reviews, there is a problem with how even when there is paper in the tray, the printer and the app reports that there is no paper in the tray. This is a cause for concern because it is a physical issue that can be fixed. A physical fix to this may be the tray or feeding technique or the way the printer scans for paper; however, it is also likely that the document has trouble in the app, causing the printer to have issues because of this. The reason that it is believed this is the case is because in many of the reviews, the customers say that when they load in paper, the wireless device they are printing on says that the printer cannot print, but when they switch to using a usb, the printer works fine. This means that the problem is not within the printer itself, but in the connection of the printer and the device.

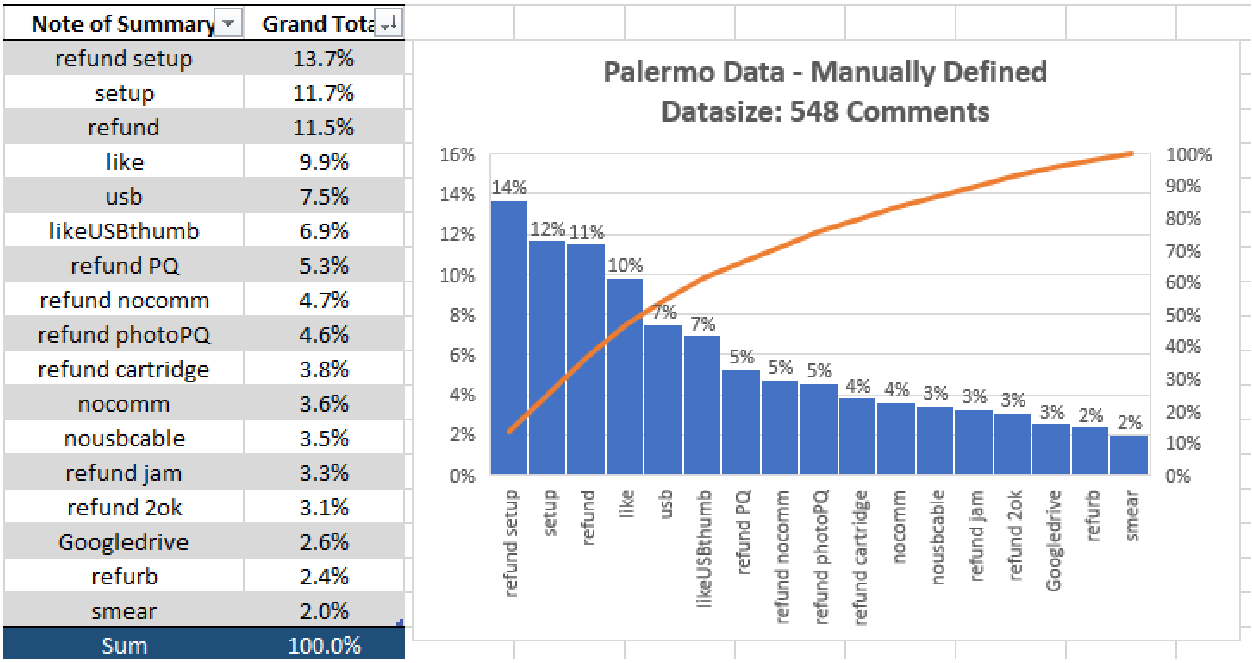
From the dataset to the right, we see that using SentenceTransformers, the largest problems for the Taccola printer are setup, refund, Instant Ink, and the printer being slow. The largest of the negative reviews is the setup, which takes up an accumulated 19% of the total reviews. It is believed that it is because the instructions of the setup are vague and do not provide troubleshooting. It says to download HP Smart and follow along with what it says. However, the HP Smart program is not consistent, and the printer can take a long time to connect. For the setup to be effective, there either needs to be a better app from HP for their printers to connect, or there needs to be an alternative connection method in case the app does not work. In the case of the Taccola, there is only one usb port, and does not come with a usb wire. HP Smart then forces the user to connect via wireless Because of this downfall, a less technologically advanced user may not be able to connect the printer. Another setup problem with the Taccola is its wifi setup. There are no actual screens on the Taccola printer for checking the wifi connection, which causes confusion or the user on which network the printer could have connected to.

From the data below, we are able to see that compared to the Palermo, the Taccola has a greater skew in data; especially in the setup area. For the online reviews based on customer satisfaction and star rating, the Taccola had a score of 62 and 3.7 respectively, while the Palermo had 73 and 4.3 respectively, compared to an overall average of 75 and 4.5 stars. An outlying problem is the high amount of people complaining about the OOP, out of paper. From the baseline of the reviews, there is a problem with how even when there is paper in the tray, the printer and the app reports that there is no paper in the tray. This is a cause for concern because it is a physical issue that can be fixed. A physical fix to this may be the tray or feeding technique or the way the printer scans for paper; however, it is also likely that the document has trouble in the app, causing the printer to have issues because of this. The reason that this is the case is because in many of the reviews, the customers say that when they load in paper, the wireless device they are printing on says that the printer cannot print, but when they switch to using a usb, the printer works fine. This means that the problem is not within the printer itself, but in the connection of the printer and the device.



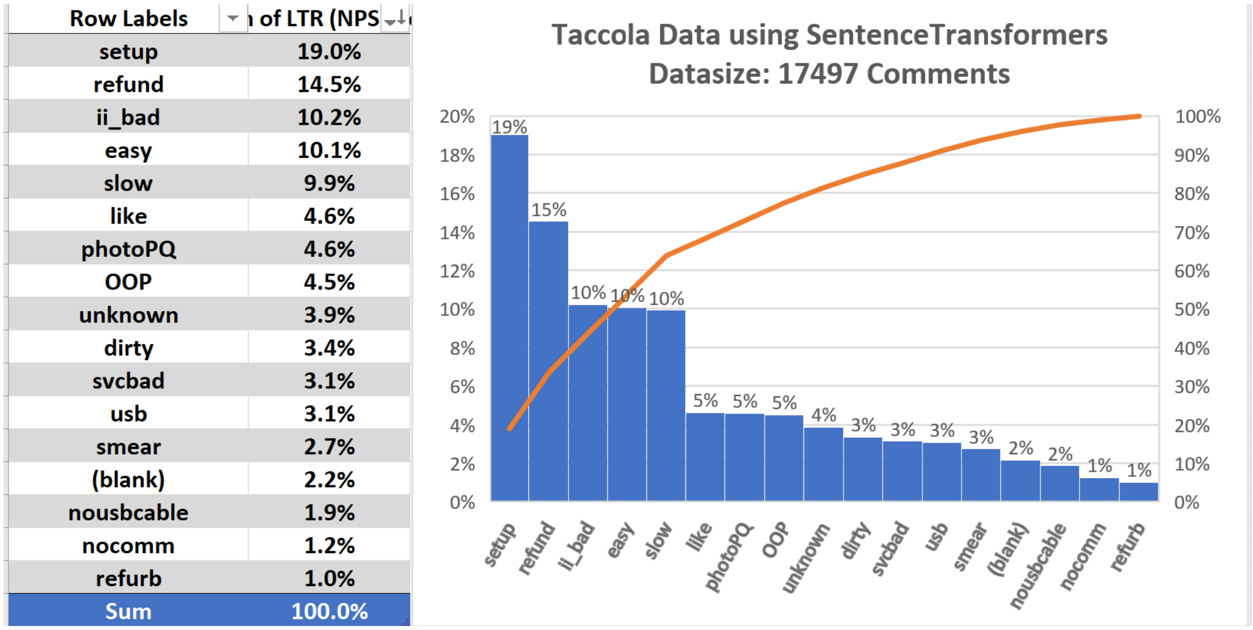
Note	1	2	3	4	5
setup	-4.8%	-9.3%	3.3%	0.8%	0.8%
refund	-7.5%	-0.9%	2.4%	1.1%	2.6%
ii_bad	-3.8%	-2.7%	3.6%	0.0%	0.0%
easy	-0.1%	-0.3%	1.1%	3.6%	4.9%
slow	-4.5%	-3.4%	1.1%	0.6%	0.3%
like	0.0%	-0.2%	0.3%	0.8%	3.4%
photoPQ	-0.9%	-1.4%	1.6%	0.3%	0.4%
OOP	-1.6%	-1.1%	1.4%	0.4%	0.2%
unknown	-0.4%	-0.9%	1.5%	0.5%	0.6%
dirty	-1.3%	-0.9%	0.7%	0.4%	0.1%
svcbad	-1.1%	-1.2%	0.6%	0.2%	0.0%
usb	-1.1%	-0.8%	0.4%	0.1%	0.6%
smear	-0.8%	-1.2%	0.4%	0.3%	0.1%
(blank)	-0.1%	-0.5%	0.9%	0.4%	0.3%
nousbcabl	-1.1%	-0.4%	0.3%	0.0%	0.0%
nocomm	-0.5%	-0.4%	0.2%	0.2%	0.0%
refurb	-0.2%	-0.1%	0.2%	0.1%	0.5%

Table 1: Data summarization for the Taccola printer, as well as top 17 reasons for comment dissatisfaction from the program.



Note of Summary	Grand Tota
refund setup	13.7%
setup	11.7%
refund	11.5%
like	9.9%
usb	7.5%
likeUSBthumb	6.9%
refund PQ	5.3%
refund nocomm	4.7%
refund photoPQ	4.6%
refund cartridge	3.8%
nocomm	3.6%
nousbcable	3.5%
refund jam	3.3%
refund 2ok	3.1%
Googledrive	2.6%
refurb	2.4%
smear	2.0%
Sum	100.0%

Table 2: Top data for customer dissatisfaction for the Palermo Printer. Data achieved through analysis for 548 comments.



Row Labels	of LTR (NPS
setup	19.0%
refund	14.5%
ii_bad	10.2%
easy	10.1%
slow	9.9%
like	4.6%
photoPQ	4.6%
OOP	4.5%
unknown	3.9%
dirty	3.4%
svcbad	3.1%
usb	3.1%
smear	2.7%
(blank)	2.2%
nousbcable	1.9%
nocomm	1.2%
refurb	1.0%
Sum	100.0%

Table 3: Top data for customer dissatisfaction for the Taccola Printer. Data achieved through analysis for 17497 comments.

Results and Conclusion

Using Sentence-Bert, SentenceTransformer, and spatial distance, a 90% accuracy rate was achieved. The data shows that most of the concerns in data lie in the setup process, starting from when the customer buys the printer, to when the printer connection does not work, linking to the customer calling customer service and ultimately returning the printer due to frustrations. The leading issues for the Taccola printer are the Setup, the Print Speed, and the Customer service for the printer itself. The reason the setup causes trouble for people is due to the vague directions of the HP Smart app. The printed instruction manual just says to follow along with the app, and if there are any troubleshooting errors, call or message the HP customer service. However, when setting up printers, there are many instances where the printer cannot be found on the device or fails to connect. This leads to the customer being frustrated with the product, and when the customer calls customer service, they often are unable to help the customer due to there being nothing they can do to assist, as there are no steps to be taken without more information and context.

Further steps to be taken could be to add in the different types of refunding in the program, so that from the refund section, we would be able to see what types of problems people are having, and why they feel like they need to refund the Taccola. With this data, we would be able to efficiently debug the most important parts of the problem with the Taccola printer by just looking at the comments with low ratings.

Acknowledgements

This work has been supported by Hewlett Packard Incorporated and the Saturday Academy ASE program. It has been advised by Jeffery Sheadel from Hewlett Packard.

References

Devlin, J. (n.d.). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. https://nlp.stanford.edu/seminar/details/jdevlin.pdf.

Devin, J., Chang, M.-W., Lee, K., &amp; Toutanova, K. (2019, May 29). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arxiv.org. https://arxiv.org/pdf/1810.04805.pdf.

Distance computations (scipy.spatial.distance) (2021). Distance computations (scipy.spatial.distance) - SciPy v1.7.1 Manual. (n.d.). https://docs.scipy.org/doc/scipy/reference/spatial.distance.html#module-scipy.spatial.distance.

Horev, R. (2018, November 17). BERT explained: State of the art language model for NLP. Medium. https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270.

Shaikh, J. (2017, October 30). *Machine learning, nlp: Text classification using SCIKIT-LEARN,Python and NLTK.* Medium. https://towardsdatascience.com/machine-learning-nlp-text-classification-using-scikit-learn-python-and-nltk-c52b92a7c73a.

Reimers, N., &amp; Gurevych, I. (2019, August 27). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. https://arxiv.org/pdf/1908.10084.pdf.