

# CMPSC 293S: Final Project Report

Andrew Lu, Ashley Wang

December 9, 2021

## 1 Objectives and Challenges

The objective of this project is to understand and reproduce the results from the paper "[Did you buy it already?](#)", *Detecting Users Purchase-State From Their Product-Related Questions*. This paper was presented on behalf of Amazon by Lital Kuchy, David Carmel, Thomas Huet, and Elad Kravi in the SIGIR 2021 conference.

The paper aims to provide a method of classification for questions asked on e-commerce websites as either pre-purchase or post-purchase questions. The motivation behind this is that knowing the purchase state of users allow e-commerce websites to present more relevant information to the user for an overall better user experience.

The challenge lies in the fact that many questions can have ambiguous purchase-state classification due to the question itself (e.g. "Does this product have a warranty?" can be both pre- and post-purchase) or due to other external variables (e.g. shipping times allow users to ask questions after time of purchase but before receiving the product). This is additionally worsened by the general complexity of our objective being a Natural Language Understanding (NLU) problem.

## 2 State-of-the-Art Techniques

For this paper, we utilize a state-of-the-art pre-trained RoBERTa<sup>1</sup> model proposed by a group of researchers from Facebook AI in 2019. The RoBERTa model is a variation of BERT that modifies the pre-training approach by (1) training the model longer, with bigger batches over more data; (2) removing the next sentence prediction objective; (3) training on longer sequences; and (4) dynamically changing the masking pattern applied to the training data. The model has been shown to get better performance on downstream tasks according to widely accepted NLU evaluation guidelines.

## 3 Dataset

The paper uses a dataset of 232,493 questions posted on Amazon.com that have a known associated purchase time. Each item in the dataset includes the item's ASIN, product title, question text, product purchase time relative to question posting time (in hours), and the pre- or post-purchase classification, based off of  $\Delta = 24$  hours. This  $\Delta$  value denotes the time after the purchase of an item where posted questions are still categorized as pre-purchase, and is used to account for product delivery time. The authors used  $\Delta = 24$  hours due to Amazon's one-day shipping guarantee and further analysis that found this value to produce the highest accuracy and F1 scores. This dataset was published along with the paper and is available on the [Registry of Open Data on AWS](#).

---

<sup>1</sup>Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. [arXiv preprint arXiv:1907.11692 \(2019\)](#).

## 4 Methodology

In this section, we briefly discuss the key methods described in the paper, and any modifications we made for its implementation.

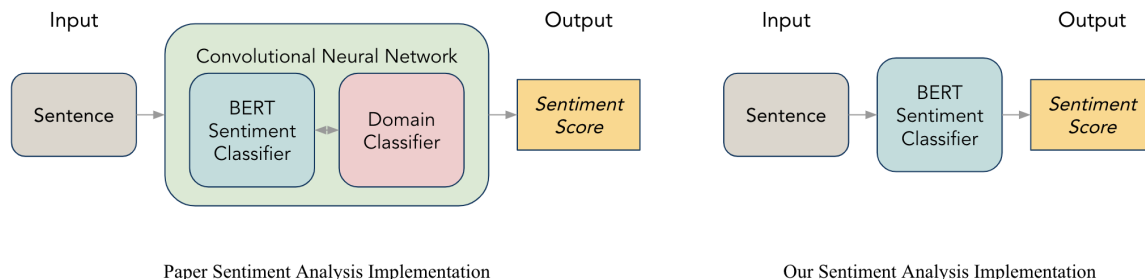
### 4.1 Manual Features

For some classification methods, the paper calculates additional manual features that are passed as input into the classifier. In this section, we discuss what some of these features are, how they are obtained in the paper, and how we plan to incorporate them into our implementation.

#### 4.1.1 Sentiment Polarity Analysis

In the paper, the researchers found that post-purchase questions tend to have more negative sentiment than pre-purchase questions. To calculate a sentiment score for the questions, they trained a sentiment analysis classifier on *Amazon Review Data*<sup>2</sup> using reviews with 1-2 stars as "negative" training points and those with 4-5 stars as "positive" training points. They additionally created a domain classifier differentiating between reviews and questions, and used both models in a convolutional neural network (CNN) classifier to ensure that sentiment scores were domain invariant.

For our project, we simplify this model by only training a sentiment classifier using BERT and disregarding the variance from the change in domain. For training the classifier, we use the same *Amazon Review Data* as the paper, and we take the direct output of the BERT classifier as our sentiment score.



#### 4.1.2 Subjectivity

Additionally, the researchers found that post-purchase questions tend to be more subjective than pre-purchase ones. They followed the same methodology as mentioned in Section 4.1.2 to create a subjectivity classifier, using reviews from *Amazon Review Data* as "subjective" training points and product descriptions as "objective" training points.

For our project, we simplify the classifier implementation to match that of the previous section, and use the same training data mentioned in the paper.

#### 4.1.3 Specificity

The researchers hypothesized that post-purchase questions would tend to be more specific due to more specific questions about the usage or product details. They calculated a specificity score based on a set of linguistic features. After graphing the distribution of the features over the purchase states, they found that question length was the most separable.

---

<sup>2</sup>Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 188–197.

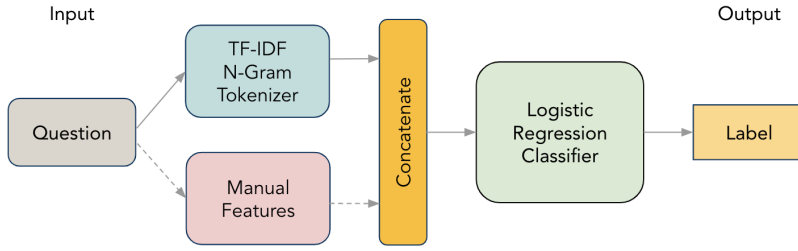
Since question length was the most separable feature out of the proposed set, we directly use the text-length as each question’s specificity score.

## 4.2 Logistic Regression

The first classification approach mentioned in the paper is a logistic regression classifier. For this method, they implemented two different models:

- $LR_Q$ : logistic regression classifier trained using the questions’ textual features only
- $LR_{Q+F}$ : logistic regression classifier trained using the questions’ textual features as well as the manual features discussed in Section 4.1.

For the textual features, they calculate a document-term frequency matrix using all unigrams and bigrams and use the matrix as an input for the classifier. If additional manual features are included, a vector of their scores are concatenated to the document-term frequency matrix.



Logistic Regression Classifier Implementation

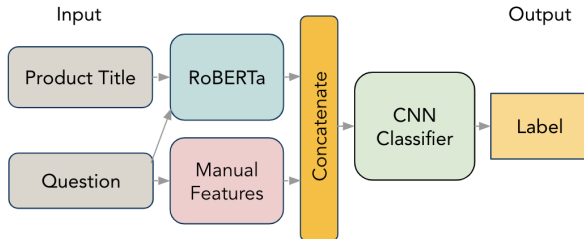
For our project, we follow the same methodology to train and test our classifier.

## 4.3 RoBERTa-Based Classification

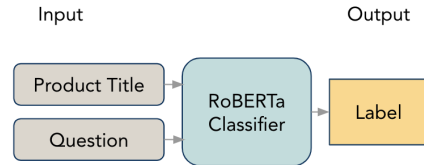
The second classification method used in the paper utilizes RoBERTa to output a vector encoding that can then be used as an input into a CNN classifier. Based on the model additional features were included for calculating the input of the classifier:

- $RoBERTa_Q$ : CNN classifier with RoBERTa encoding on the question text only.
- $RoBERTa_{QT}$ : CNN classifier with RoBERTa encoding on the question text and product title.
- $RoBERTa_{QT+F}$ : CNN classifier with RoBERTa encoding on the question text and product title concatenated with the manual features discussed in Section 4.1.

For our project, we simplify this model to use RoBERTa as the classifier directly (instead of using its output as an input to a CNN classifier). Since we no longer separate the encoding and classification steps, we only implement a variation of  $RoBERTa_Q$  and  $RoBERTa_{QT}$  as we have no input vector to concatenate the manual features to.



Paper RoBERTa-Based Classification Implementation



Our RoBERTa Classifier Implementation

## 5 Results

For the paper, the authors primarily use the accuracy and average F1 scores over both purchase states as metrics for evaluation. For our project, we use the same metrics so that we can compare the similarity of our results.

For our metrics, we calculate the accuracy and F1 scores using 30% of the dataset as our test set. We divide the dataset into a training and testing set using a consistent random seed so that all models have the same training and testing set.

| Model          | Paper Results |                   | Our Results |                   |
|----------------|---------------|-------------------|-------------|-------------------|
|                | Acc           | F1 <sub>avg</sub> | Acc         | F1 <sub>avg</sub> |
| $LR_Q$         | 0.741         | 0.729             | 0.773       | 0.762             |
| $LR_{Q+F}$     | 0.756         | 0.744             | 0.775       | 0.763             |
| $RoBERTa_Q$    | 0.801         | 0.790             | 0.791       | 0.780             |
| $RoBERTa_{QT}$ | 0.805         | 0.795             | 0.782       | 0.770             |

For our logistic regression classifier using only question text ( $LR_Q$ ), our accuracy and F1 scores were much higher than those obtained in the paper. Since we followed the paper’s methodology as stated, we believe the discrepancy is simply due to the seed that we chose for our train and test set division.

Further adding in the three manual features ( $LR_{Q+F}$ ) also showed an improvement over the paper results, but this improvement was not as significant as that shown in the paper (they had an improvement of 0.015 accuracy versus our 0.002 improvement). This was not surprising, as the paper reports that the "Question and Expected Answer Types" manual feature showed the highest accuracy improvement of the manual features that they tested. For our classifier, the "Sentiment" and "Subjectivity" manual features brought the most improvement to our accuracy, while "Specificity" had little effect. This was consistent with the findings in the paper as well.

For our RoBERTa classifier using only question text ( $RoBERTa_Q$ ), our results were similar, although slightly lower, than those presented in the paper. As we don’t know the exact implementation and model training parameters that the authors used, we believe that differences from their results can be attributed to the paper authors being able to try different configurations for training the classifier (e.g. different batch sizes, more epochs, and different optimizers) on larger, more powerful machines.

Our RoBERTa classifier with question and title text ( $RoBERTa_{QT}$ ), however, showed a decrease in accuracy which was inconsistent with the results from the paper. Since the authors did not specify how they incorporated the product’s title into the RoBERTa encoder, we simply appended the product title to the question text, which likely caused questions to seem more neutral, instead of the original goal of adding context. Despite the lower accuracy than the paper results, the performance of the  $RoBERTa_{QT}$  classifier was still better than both logistic regression classifiers, leading us to conclude that overall RoBERTa is a better classifier for this problem.

## 6 Our Efforts

Given that the code from the original paper was not released, we had to implement many of the methods from the paper by ourselves. This involved reading many of the cited sources from the original paper and determining if the methodology was within the scope of the class or not. Since they utilized many machine learning concepts, we decided to focus our attention on the applications of BERT and its variations.

For this project, we researched different methods to vectorize a document vector and implement a logistic regression model. Ultimately we decided on the SciKit Learn [TfidfVectorizer](#) and [LogisticRegression](#) model since they were fairly straightforward and easy-to-use.

Similarly, we also looked into different methods to use BERT as a classifier. For this portion of the project we decided to use the [Tensorflow BERT](#) model since examples of its application as a classifier were available.

In trying to implement our RoBERTa classifier models, we found that since RoBERTa was fairly new, the documentation for running a pre-trained RoBERTa classifier was lacking, and we had to experiment with many of different RoBERTa APIs. This is in contrast to the base BERT model which has a fair amount of documentation and instruction on how to build a classifier. After experimenting with a few different libraries, we decided to use the [RoBERTa transformer published by Uekjae Jeong](#) since we could utilize a similar method the one we used for sentiment analysis in manual features.

In addition the lack of well-documented RoBERTa models, we ran into many road-blocks due to hardware limitations, as fine-tuning our classifier took up to 150 hours to train without a GPU. To resolve this, we tried utilizing different libraries that could be trained using a GPU; however, those libraries often resulted in Out-Of-Memory errors or documentation was lacking. Thus, for the sake of obtaining comprehensible results, we chose to persist with our original implementation.

## 6.1 Relation to Class Material

For our logistic regression models, we looked into TF-IDF vectorization for unigrams and bigrams of our strings, which was discussed in the [Document Retrieval Models](#) lecture and was also used in the [Lucene](#) document ranking experiments. While the TF-IDF scores were primarily discussed within the context of ranking for document retrieval, in our project we applied it mostly as a measure of similarity between documents and used that measure for classification in a logistic regression model.

Additionally, we used a BERT classifier for our manual features and the optimized RoBERTa classifier for purchase-state classification. These applications of BERT were discussed in the [BERT: Contextual Representations for Better Text Understanding](#) lecture. Furthermore, to understand the difference between the base model of BERT and RoBERTa, we had to delve deeper into the difference in the pre-training methods of BERT and RoBERTa, which was discussed in the [Pretrained Transformers for Text Ranking: BERT and Beyond](#) lecture.

## 7 Conclusion

In conclusion, we found that after recreating simplified versions of the methods introduced in the paper, we were able to produce similar results and any discrepancies could be attributed to the simplification or ambiguity of the original methodology. Overall we found that RoBERTa is the superior classification/encoding method (compared to logistic regression using TF-IDF as measures of similarity) when it comes to problems relating to NLU. This is shown in our results as methods using RoBERTa as a classifier had significantly better accuracy and F1 scores than those produced by our logistic regression models.

Through this project, we were able to utilize similarity scores typically used for ranking (i.e. TF-IDF vectorization) in another context (i.e. logistic regression classification). Additionally, we explored applications of the BERT model introduced in class, and learn more about how BERT and its variations can be used both as a classifier and as an encoder.

# “Did You Buy It Already?”

Detecting Users Purchase-State From  
Their Product-Related Questions




Lital Kuchy, David Carmel, Thomas Huet & Elad Kravi  
SIGIR 2021

Presented by Ashley Wang & Andrew Lu

# Motivation

- eCommerce websites have Q&A platforms where users can ask questions about products for sale
- Certain question types are asked more frequently depending on the user's purchase stage (pre- or post-purchase)
- Knowing purchase stage allows the user to be presented with questions containing more relevant information

# Motivation

| Product                             | Image   | Pre-purchase questions  | Post-purchase questions   |
|-------------------------------------|---|---|---|
| Ring Video Doorbell (1st Gen)       |  | <ul style="list-style-type: none"><li>• How long does the battery last?</li><li>• Does this work with Google Home?</li><li>• What is the difference between ring video door and ring video doorbell 2</li></ul> | <ul style="list-style-type: none"><li>• How do I enable live view?</li><li>• When charging, how do you turn it off?</li><li>• Why is my Ring doorbell battery losing its charge while connected to wires?</li></ul> |
| eufy by Anker, Robot Vacuum Cleaner |  | <ul style="list-style-type: none"><li>• Does it work on carpets?</li><li>• Does it work on pet hair?</li><li>• Is this 220V?</li></ul>  | <ul style="list-style-type: none"><li>• Can I buy a replacement remote control?</li><li>• It says not to use on dark floors. Why? How dark?</li><li>• How long does it take to get a full charge?</li></ul>         |
| Oculus Quest VR Gaming Headset      |  | <ul style="list-style-type: none"><li>• When does shipping start?</li><li>• Can you stream to a roku?</li><li>• Can you play Minecraft</li></ul>  | <ul style="list-style-type: none"><li>• Why does it take so long to ship?</li><li>• Can u upgrade the memory?</li><li>• Is there a way to cast what you're seeing on a TV or monitor?"</li></ul>                    |



# Motivation

- eCommerce websites have Q&A platforms where users can ask questions about products for sale
- Certain question types are asked more frequently depending on the user's purchase stage (pre- or post- purchase)
- Knowing purchase stage allows the user to be presented with questions containing more relevant information

However, for **74%** of product questions on Amazon.com, the actual **purchase time is unknown**.

# Problem Definition

Given a product related question, can we determine whether the question was asked ***before*** or ***after*** the product was purchased?

# What are Pre- and Post-Purchase Questions?

- **Pre-purchase:** before or within  $\Delta$  hours of purchase
- **Post-purchase:** otherwise
- $\Delta$  accounts for time after purchase but before receipt
- Use  $\Delta=24$  hours because Amazon provides one-day shipping

# Dataset

232,493 product questions from Amazon.com containing:

- Item ASIN
- Item title
- Question text
- Question post time relative to purchase time, in hours
- Corresponding purchase-stage classification

Dataset is released on the [Registry of Open Data on AWS](#) along with this paper's publication.

# Classification Methods

- Human Classification
- Logistic Regression
  - Textual features only ( $LR_Q$ )
  - Textual features with manual features ( $LR_{Q+F}$ )
- RoBERTa-Based Classifier
  - Question text only ( $RoBERTa_Q$ )
  - Question text and product title ( $RoBERTa_{QT}$ )
  - Question text, product title, and manual features ( $RoBERTa_{QT+F}$ )

# Manual Features

- Sentiment Polarity Analysis
- Subjectivity
- Specificity
- Parts-of-Speech (POS) Patterns
- Question and Expected Answer Type

# Sentiment Polarity Analysis

- Post-purchase questions express more negative sentiment

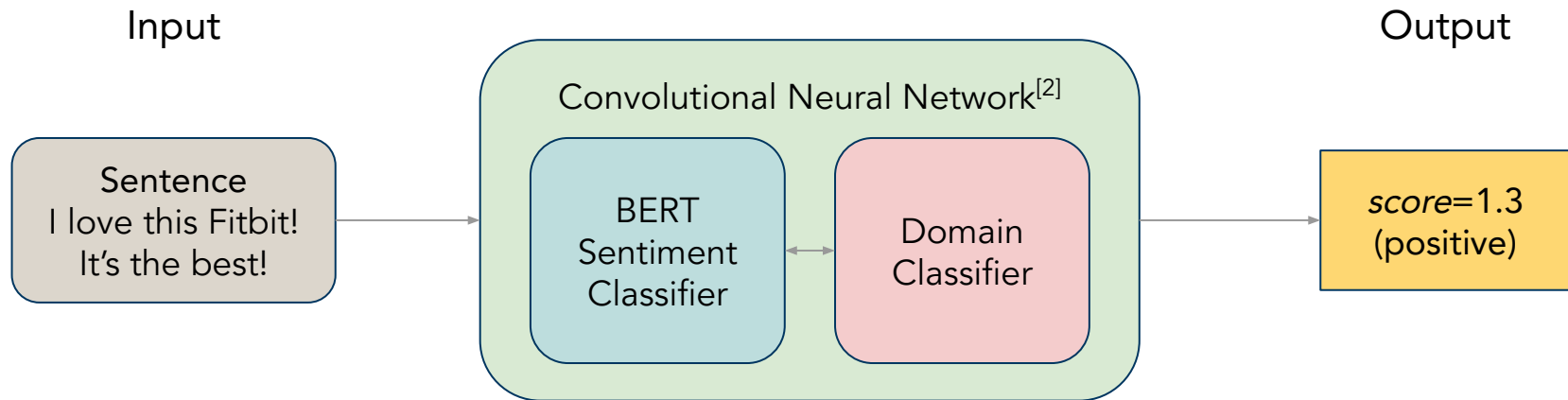
| Label | Sentiment |          |
|-------|-----------|----------|
|       | Negative  | Positive |
| Pre   | 44%       | 56%      |
| Post  | 60%       | 40%      |

- BERT Classifier using Amazon Review Data<sup>[1]</sup>
  - Includes: product ASIN, reviewer ID and name, helpfulness, review, summary, overall rating, review time
  - Negative: 1-2 stars | Positive: 4-5 stars

[1] Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 188–197. [Link](#).

# Sentiment Polarity Analysis

- Domain classifier to predict if input is question or review
- Optimized to decrease loss of sentiment classifier



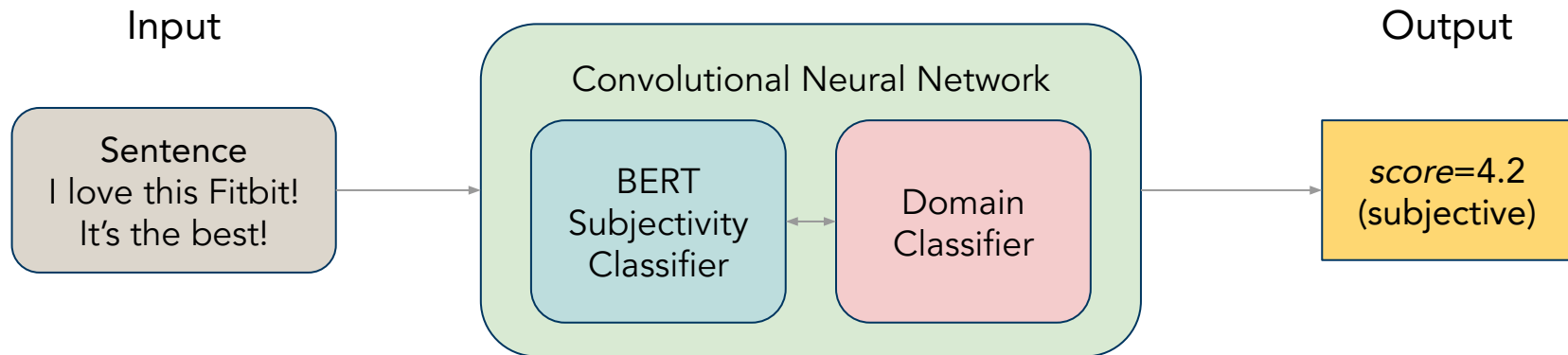
[2] Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*. PMLR, 1180–1189. [Link](#).



# Subjectivity

- Post-purchase questions are more subjective
- Similar method to sentiment analysis (BERT + domain classifier)
  - Amazon Review Data & Product Metadata
  - Subjective: Review | Objective: Product descriptions

| Label | Subjectivity |           |
|-------|--------------|-----------|
|       | Subjective   | Objective |
| Pre   | 50%          | 50%       |
| Post  | 74%          | 26%       |



# Specificity

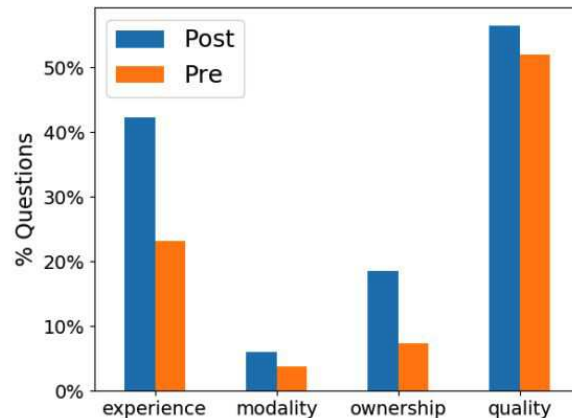
- Hypothesize that post-purchase questions are more specific
- Linguistic Feature Set<sup>[3]</sup>
  - Text-length
  - Number of occurring numbers
  - Capital letters
  - Non-alphanumeric symbols
- Measure distribution of features over purchase state
- Text-length was most separable
  - Pre-purchase: 11.29 words | Post-purchase: 14.67 words

[3] Junyi Jessy Li and Ani Nenkova. 2015. Fast and accurate prediction of sentence specificity. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*. [Link](#).

# Parts-Of-Speech Patterns

- Used NLTK framework to label questions with POS tags<sup>[4]</sup>

| Tag Type   | Definition                           | Examples                                      |
|------------|--------------------------------------|---|
| OWNERSHIP  | Possessive Pronouns                  | "my first", "his second"                      |
| QUALITY    | Comparative + Superlative Adjectives | "best", "bigger"                              |
| EXPERIENCE | Past-tense Verbs, Past Participles   | "owned", "driven"                             |
| MODALITY   | Hypotheticals                        | "I can", "you will",<br>"If I could, I would" |

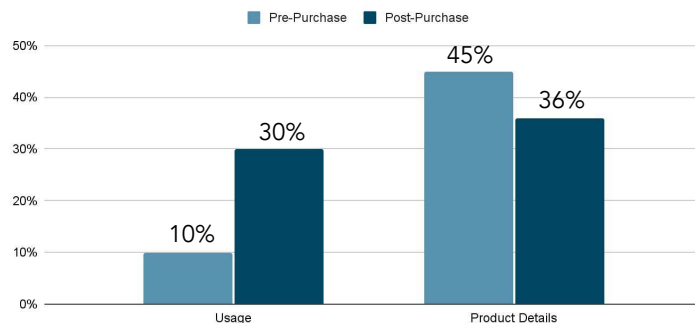


[4] Mehedi Hasan, Alexander Kotov, Aravind Mohan, Shiyong Lu, and Paul M Stieg. 2016. Feedback or research: separating pre-purchase from post-purchase consumer reviews. In *European Conference on Information Retrieval*. Springer, 682– 688. [Link](#).

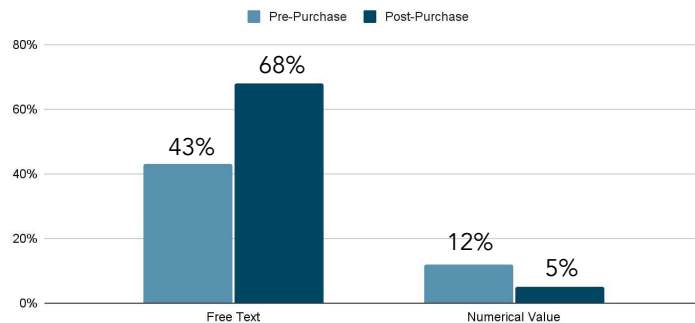
# Question and Expected Answer Type

- Multiclass classification
- Question Types:
  - product details
  - product relation
  - usage
  - product search
  - general knowledge
  - subjective opinion
- Answer Types:
  - yes/no
  - date/time
  - numeric value
  - free text

Question Type Classification



Expected Answer Type Classification



# Logistic Regression

- Generalized Linear Model used for binary classification
- Estimator outputs probability  $p$  of input belonging to a class

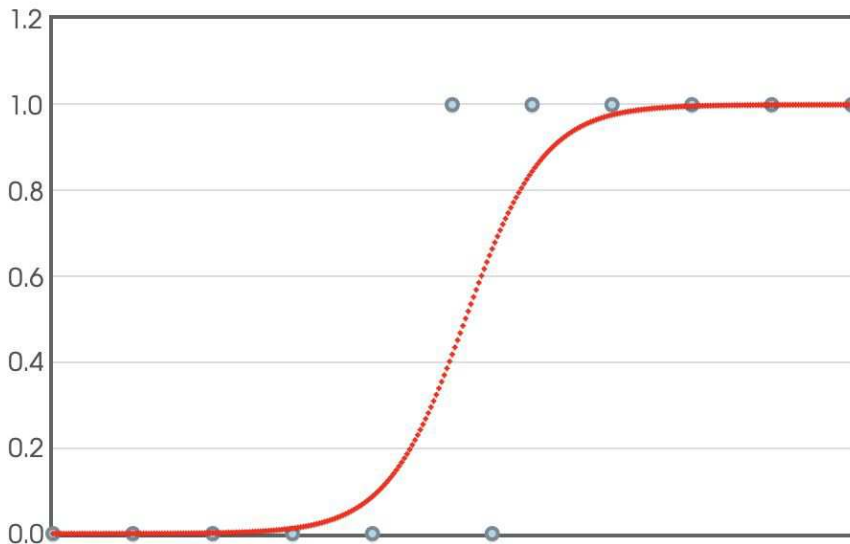
Estimator: 
$$p(x) = \frac{e^{\ell(x)}}{e^{\ell(x)} + 1}$$

Where:

Given input  $x = \langle x_1, x_2, \dots, x_n \rangle$ ,

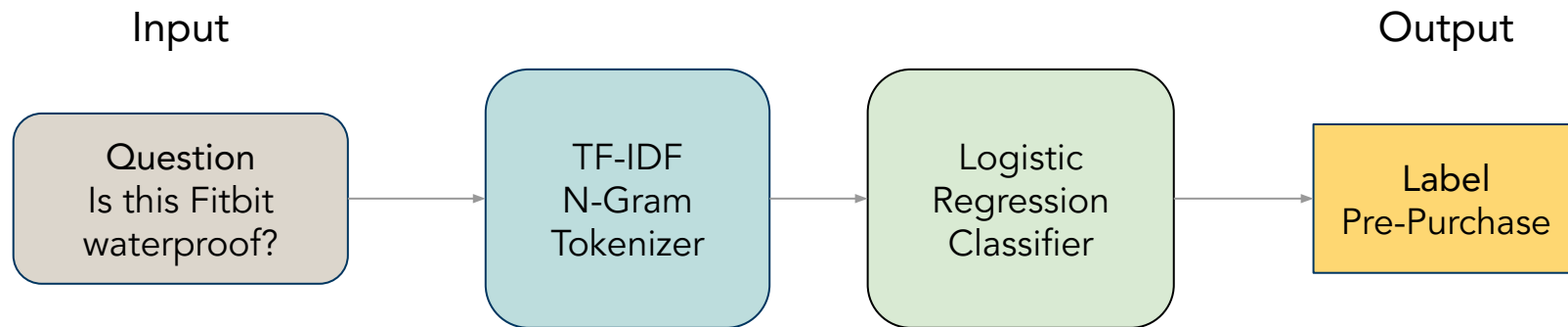
$$\ell(x) = a_0 + a_1x_1 + a_2x_2 + \dots + a_nx_n$$

$a_i$  : coefficients determined from MLE

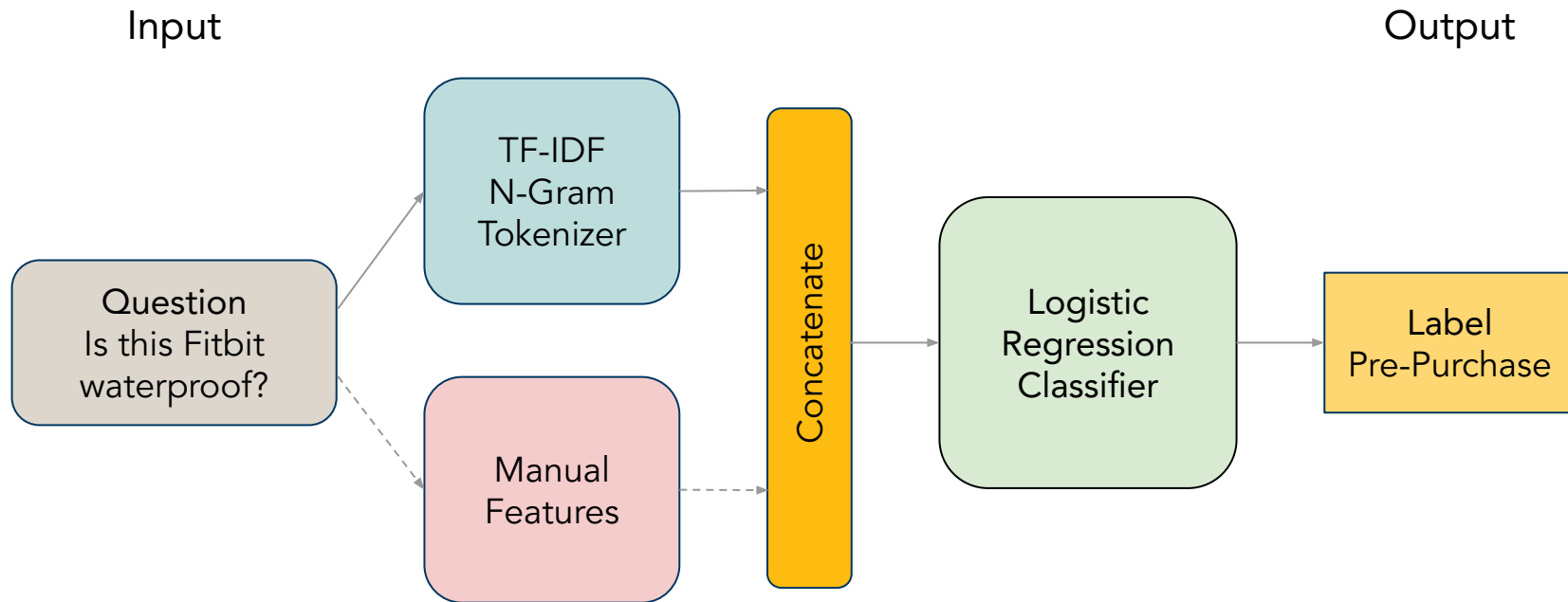


# Logistic Regression: N-Gram Tokenization

- Create set of all unigram and bigrams ( $n=1, n=2$ )
  - N-grams: all consecutive  $n$  words as single term
- Calculate TF-IDF to make document-term frequency matrix



# Logistic Regression With Manual Features



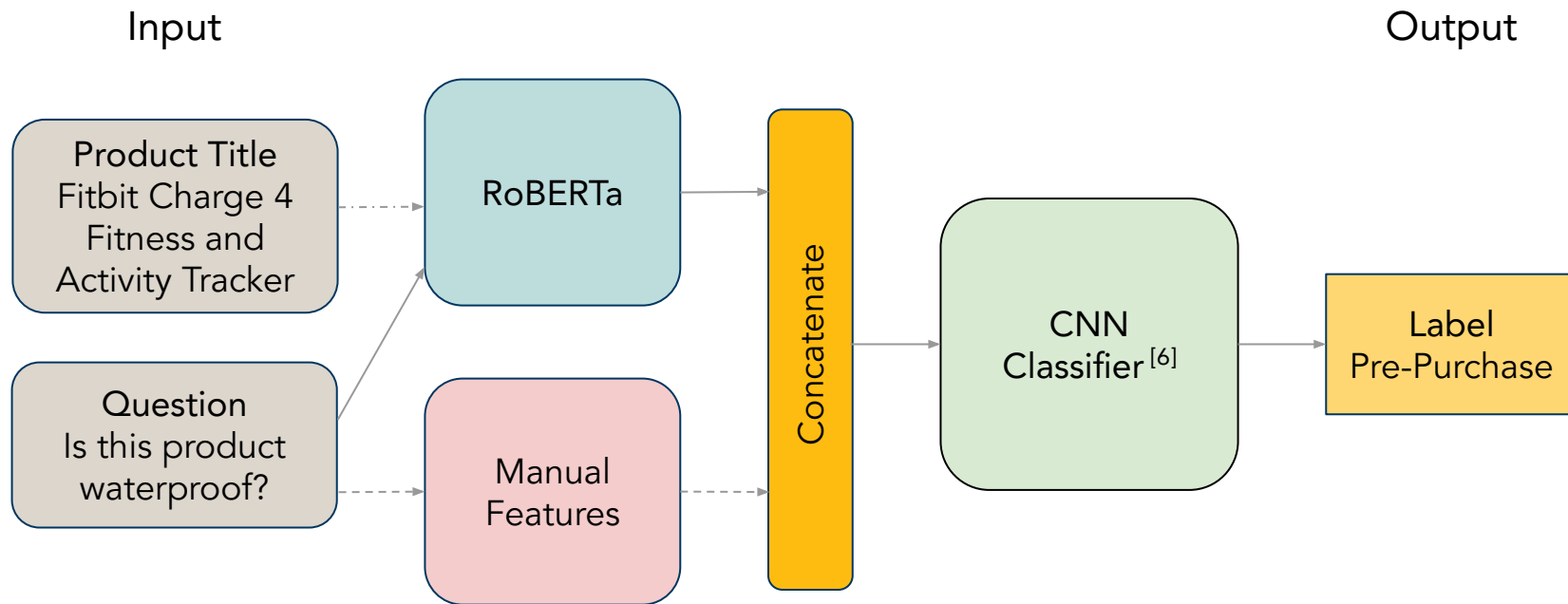
# RoBERTa Model

- BERT: word embeddings for better language understanding
  - Pre-training includes Masked Language Model (predicting a single word) and Next Sentence Prediction
- RoBERTa: Robustly Optimized BERT Approach<sup>[5]</sup>
  - Improved training algorithm utilizing more data and training time
  - Removed next sentence prediction objective
  - Better performance for downstream tasks

[5] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019). [Link](#).



# RoBERTa-Based Classification



[6] Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* (2014). [Link](#).

# Overall Accuracy & F1 Results

Accuracy and F1 results of the different classification methods, with  $\Delta = 24$  hours

| Model                   | Acc.           | F1             |
|-------------------------|----------------|----------------|
| $\text{LR}_Q$           | 0.741          | 0.729          |
| $\text{LR}_{Q+F}$       | 0.756          | 0.744          |
| $\text{RoBERTa}_Q$      | 0.801*         | 0.790*         |
| $\text{RoBERTa}_{QT}$   | <b>0.805**</b> | <b>0.795**</b> |
| $\text{RoBERTa}_{QT+F}$ | 0.804          | 0.793          |
| Human                   | 0.756          | 0.725          |

# Pre- vs. Post-Purchase Precision & Recall

Classification results for RoBERTa<sub>Q</sub> and RoBERTa<sub>QT</sub> separated into pre- and post-purchase questions, with  $\Delta = 24$  hours

| Method                | Pre          |              |              | Post         |              |              |
|-----------------------|--------------|--------------|--------------|--------------|--------------|--------------|
|                       | Prec.        | Recall       | F1           | Prec.        | Recall       | F1           |
| Human                 | 0.720        | <b>0.943</b> | 0.817        | <b>0.866</b> | 0.500        | 0.634        |
| RoBERTa <sub>Q</sub>  | 0.789        | 0.895        | 0.839        | 0.825        | 0.673        | 0.7419       |
| RoBERTa <sub>QT</sub> | <b>0.797</b> | 0.887        | <b>0.840</b> | 0.819        | <b>0.692</b> | <b>0.750</b> |

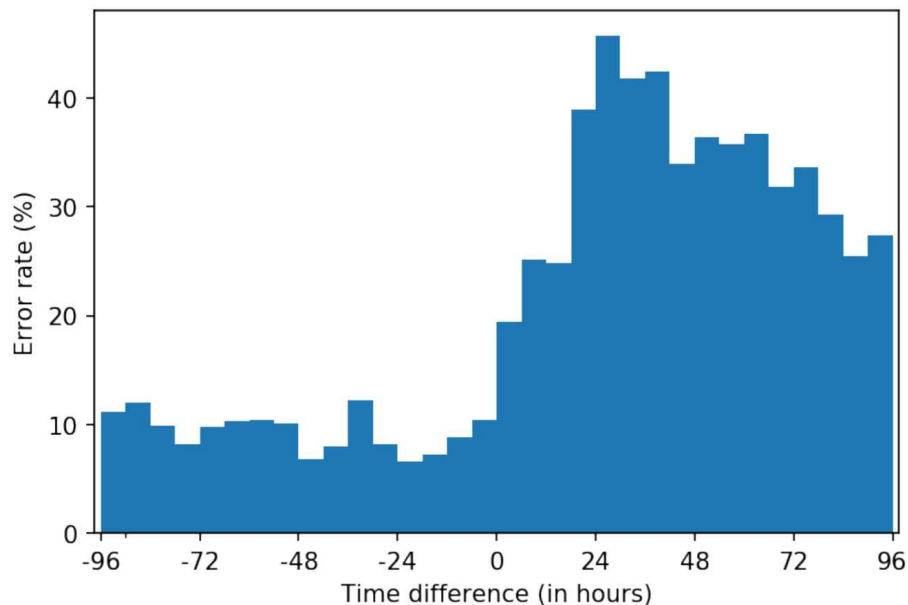
# Modifying $\Delta$ Values

Accuracy and F1 results of RoBERTa<sub>QT</sub> with various  $\Delta$  values (labels)

| Label method        | High-confidence |       | $\Delta$ -based |       |
|---------------------|-----------------|-------|-----------------|-------|
|                     | Acc.            | F1    | Acc.            | F1    |
| 0 <i>H</i> -labels  | 0.811           | 0.806 | 0.771           | 0.769 |
| 12 <i>H</i> -labels | 0.817           | 0.809 | 0.797           | 0.788 |
| 24 <i>H</i> -labels | 0.817           | 0.809 | 0.805           | 0.795 |
| 36 <i>H</i> -labels | 0.816           | 0.805 | 0.803           | 0.789 |
| 48 <i>H</i> -labels | 0.811           | 0.799 | 0.800           | 0.780 |

# Error Analysis

Error rate of RoBERTa<sub>QT</sub> classifier based on post and purchase time difference



# Error Analysis

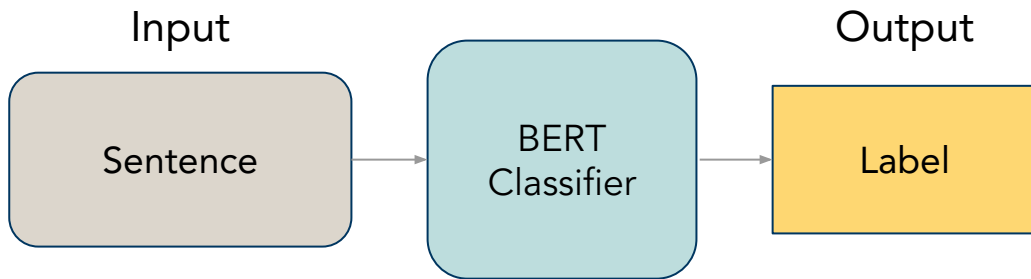
## Classification Errors

- Ambiguity
- Before Delivery
- Cynicism
- Verification

| Question  | Label | Human | RoBERTa <sub>QT</sub> |
|---|-------|-------|-----------------------|
| "What type of songs can be preloaded?"                          | Pre   | ✓     | ✗                     |
| "How do I know how much space is left on the internal storage?" | Post  | ✓     | ✗                     |
| "What is the return policy for this item?"                      | Pre   | ✗     | ✓                     |
| "Is there a warranty on this watch?"                            | Post  | ✗     | ✓                     |
| "How can I get a replacement key?"                              | Pre   | ✗     | ✗                     |
| "Do the knives actually cut?"                                   | Post  | ✗     | ✗                     |

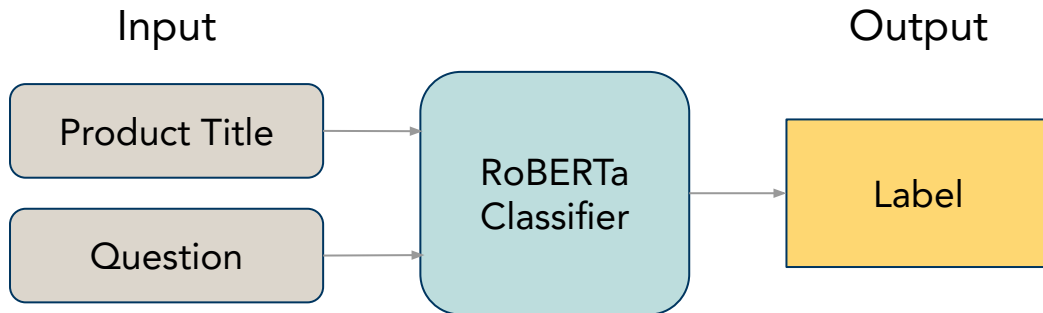
# Our Project Plan

- Recreate simplified versions of classification models and try to produce similar results
- Manual Features
  - Sentiment & Subjectivity Analysis: BERT Classifier only
  - Specificity: Text-length
  - POS Tags & Question/Expected Answer Type: Ignore



# Our Project Plan

- Logistic Regression Models
  - TF-IDF Vectorization + Logistic Regression: [SciKit Learn](#)
- RoBERTa Classification
  - Use RoBERTa for classification without CNN
  - RoBERTa Model: [PyTorch RoBERTa by Facebook AI](#)





# References

Lital Kuchy, David Carmel, Thomas Huet, and Elad Kravi. 2021. "Did you buy it already?", Detecting Users Purchase-State From Their Product-Related Questions. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21), July 11–15, 2021, Virtual Event, Canada*. ACM, New York, NY, USA, 10 pages.

<https://doi.org/10.1145/3404835.3462940>

- [Pre- and post-purchase product questions - AWS Open Data](#)
- [Amazon Product Data \(Manual Feature Dataset\)](#)