# I Can't believe you ask me that! Sentiment analysis model converted to questions.

Daniel Shterenberg, Hadar Pur,
Rotem Feinblat

Submitted as final project report for the NLP course, IDC, 2021

## 1 Introduction

The task of a question answering (QA) system is to automatically answer questions asked by humans, expressed in a natural language. In recent years, platforms, where QA systems are applicable, have emerged on the web and gained in popularity. Related to the answering of complex questions, with subjective or ambiguous answers, there is a growing interest in understanding how semantic features can be utilized further to enhance the capability of the systems.

Sentiment analysis is a natural language processing method used to identify and study the subjective information in texts. It is widely used for tasks such as interpreting reviews, analyzing social media, marketing, and finding customer opinions about products. The binary classification task of labeling a document as either positive or negative is called sentiment polarity classification, or just polarity classification.

In this project, we investigated how question answering model can predict sentiment polarity classification only from the question itself, without any information that contain sentiment in the question.

In addition to the QA model, an additional baseline model, which consider sentiment, was implemented and used as a reference point for evaluation. The QA model were trained and tested on the popular Stanford Question Answering model by using Tweets sentiment data-set .

### 1.1 Related Works

During the project we inspired by a few sources:

1. NLP Progress as main web information source, SQuAD 2.0 data-set in particular [2].

2. Related works and repositories for QA and SA models [3][5].

# 2 Solution

## 2.1 General approach

The main problem of this task is to convert sentiment analysis data-set and adapt it into a question-answering format that would match to the question-answering SQUAD model. First, we chose a QA model based on the BERT language model. We implemented the SQUAD model and evaluated it, we chose the SA data-set that contain tweets from twitter which were classified to positive and negative feelings. The second step was to adapt the sentiment data to the QA format. We transformed the data to the SQUAD format and then we trained the QA model with the edited sentiment data and compare the results from QA model to Sentiment model.

## 2.2 Data-sets

The Stanford Question Answering Data-set (SQuAD) is a reading comprehension data-set, consisting of questions posed by crowdworkers on a set of Wikipedia articles. The answer to every question is a segment of text (a span) from the corresponding reading passage. Recently, SQuAD 2.0 has been released, which includes unanswerable questions. The public leaderboard is available on the SQuAD website.

Figure 1: Squad data-set

| | title | context | question | id | answer.start | answer.text | impossible |
|---|---|---|---|---|---|---|---|
| 0 | Beyoncé | Beyoncé Giselle Knowles-Carter (/biːˈjɒnseɪ/ b... | When did Beyonce start becoming popular? | 56be85543aeaaa14008c9063 | [269] | [in the late 1990s] | False |
| 1 | Beyoncé | Beyoncé Giselle Knowles-Carter (/biːˈjɒnseɪ/ b... | What areas did Beyonce compete in when she was... | 56be85543aeaaa14008c9065 | [207] | [singing and dancing] | False |
| 2 | Beyoncé | Beyoncé Giselle Knowles-Carter (/biːˈjɒnseɪ/ b... | When did Beyonce leave Destiny's Child and bec... | 56be85543aeaaa14008c9066 | [526] | [2003] | False |
| 3 | Beyoncé | Beyoncé Giselle Knowles-Carter (/biːˈjɒnseɪ/ b... | In what city and state did Beyonce grow up? | 56bf6b0f3aeaaa14008c9601 | [166] | [Houston, Texas] | False |
| 4 | Beyoncé | Beyoncé Giselle Knowles-Carter (/biːˈjɒnseɪ/ b... | In which decade did Beyonce become famous? | 56bf6b0f3aeaaa14008c9602 | [276] | [late 1990s] | False |

The Sentiment140 data-set contains 1,600,000 tweets which were extracted using Twitter's API . The tweets have been annotated (0 = negative, 4 = positive) and they can be used to detect sentiment. It contains the following 6 fields:

1. Target: The polarity of the tweet (0=negative, 2=neutral, 4=positive).
2. Ids: The id of the tweet (2087).
3. Date: The date of the tweet (Sat May 16 23:58:44 UTC 2009).
4. Flag: The query (lyx). If there is no query, then this value is NO QUERY.
5. Text: The text of the tweet (lyx is cool).

Figure 2: Tweets data-set

| | target | ids | date | flag | user | text |
|---|---|---|---|---|---|---|
| 0 | 0 | 1467810369 | Mon Apr 06 22:19:45 PDT 2009 | NO_QUERY | _TheSpecialOne_ | @switchfoot http://twitpic.com/2y1zl - Awww, t... |
| 1 | 0 | 1467810672 | Mon Apr 06 22:19:49 PDT 2009 | NO_QUERY | scotthamilton | is upset that he can't update his Facebook by ... |
| 2 | 0 | 1467810917 | Mon Apr 06 22:19:53 PDT 2009 | NO_QUERY | mattycus | @Kenichan I dived many times for the ball. Man... |
| 3 | 0 | 1467811184 | Mon Apr 06 22:19:57 PDT 2009 | NO_QUERY | ElleCTF | my whole body feels itchy and like its on fire |
| 4 | 0 | 1467811193 | Mon Apr 06 22:19:57 PDT 2009 | NO_QUERY | Karoli | @nationwideclass no, it's not behaving at all.... |

## 2.3 Design

The original labeled data-set (Positive/Negative) contains 1.6M tweets which equally distributed to positive and negative (80k each). We shrank the data-set (due to computational resources) to 20,000 tweets - 10,000 positive and 10,000 negative. In order to match the sentiment analysis data-set to the QA model we did few steps to process the data.

First, we removed all the mentions from the tweets in order to avoid unrecognized words and unbalanced word-vector (figure 3).

Figure 3: Tweets without mention

| | target | ids | date | flag | user | text |
|---|---|---|---|---|---|---|
| 1 | 0 | 1467810672 | Mon Apr 06 22:19:49 PDT 2009 | NO_QUERY | scotthamilton | is upset that he can't update his Facebook by ... |
| 3 | 0 | 1467811184 | Mon Apr 06 22:19:57 PDT 2009 | NO_QUERY | ElleCTF | my whole body feels itchy and like its on fire |
| 6 | 0 | 1467811592 | Mon Apr 06 22:20:03 PDT 2009 | NO_QUERY | mybirch | Need a hug |
| 10 | 0 | 1467812416 | Mon Apr 06 22:20:16 PDT 2009 | NO_QUERY | erinx3leannexo | spring break in plain city... it's snowing |
| 11 | 0 | 1467812579 | Mon Apr 06 22:20:17 PDT 2009 | NO_QUERY | pardonlauren | I just re-pierced my ears |

Second, we matched the processed data to the QA format by adding a 'default' question prefix (figure 4).

Figure 4: SA data-set after modification to QA

| | context | question | answer |
|---|---|---|---|
| 1 | is upset that he can't update his Facebook by ... | would you feel positively about is upset that ... | 0 |
| 3 | my whole body feels itchy and like its on fire | would you feel positively about my whole body ... | 0 |
| 6 | Need a hug | would you feel positively about Need a hug ? | 0 |
| 10 | spring break in plain city... it's snowing | would you feel positively about spring break i... | 0 |
| 11 | I just re-pierced my ears | would you feel positively about I just re-pier... | 0 |

And lastly, we divided the modified data to 80% train set and 20% test set, trained the two models on Google Colab for few hours, evaluated the trained models and compared the results.

# 3 Experimental results

The following graph shows the train loss VS. the epochs for QA and SA models.

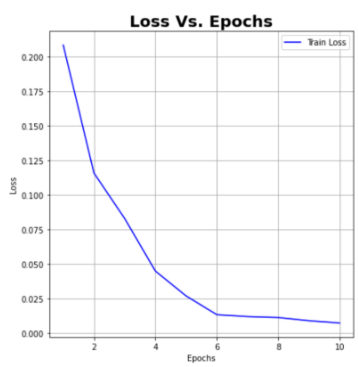


Figure 5: QA train loss VS. epoch

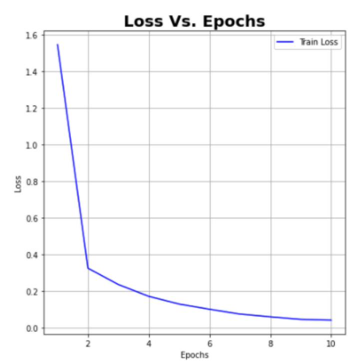


Figure 6: SA train loss VS. epoch

As we can see in fig 5 and fig 6 the train loss decrease with the epoch. The accuracy of the QA model is 91.83% and the accuracy of SA model is 83.35%.

If we will compare the 2 graphs side by side, we can see the difference between those 2 models, as we can see in fig 7.

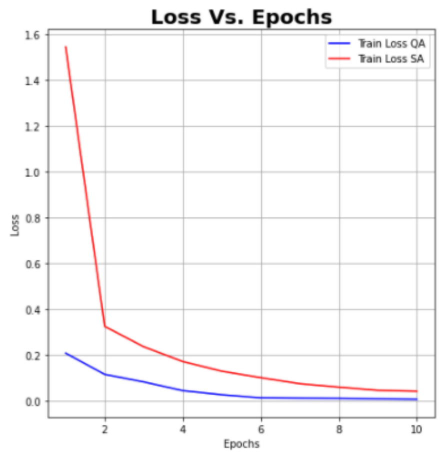


Figure 7: SA and QA train loss VS. epoch

# 4 Discussion

The results show that there was a faster convergence for the QA model compared to the SA model. It can be an indication for over-fitting in the QA model (fig 7). By the fact that the accuracy of QA was higher SA it can be concluded that the sentiment features most don't improved the ability to distinguish if the feeling is positive or negative.

## 4.1 Challenges

- A main challenge in our task was to find data-set that can be modified to QA format. At first we wanted to work with the IMDB data-set, but once we understood that the text is pretty long and that its going to be complicated to modify its context to a question format, we decided to work with tweets which easier to modify into questions.
- The original tweets data-set was huge and required a lot of computational resources. Hence, we decided to shrink the data-set and work with smaller data-set.
- The data-transformation. We were not sure how to transform the SA data into a QA data. After few trials we decided to try the naive approach.

- During our experiments, we noticed that the loss converges too fast. At first we used 10 epochs, and we noticed that it converges for the final loss value after 4-5 epochs. So if we had more resources we would have checked the affect of the learning rate and how it improves the performance.

### 4.2 Conclusions and Future work

The objective of this project was to investigate how a QA model can predict sentiment from the question itself and without using sentiment 'pre-knowledge'. The results showed that if we are taking an appropriate data-set which can modified to the desired format, we can train a QA model to answer on 'sentimental' questions with a relative success. For future work, there are several factors that need to be considered ahead:

- The model should be trained and evaluated on a larger data set.
- If using this specific data set, we need to take in consideration the context of the question and combine it with sentiment-related information, in order to improve the detection.

## 5 Code

`https://drive.google.com/file/d/1WP428AJajw1QeBKuJXz_-BRh7xEjDH3q/view?usp=sharing`

## References


1. Twitter Data-set: `http://help.sentiment140.com/for-students`
2. NLP Progress: `http://nlpprogress.com/`
3. Repository that contains an implementation of the question-answering system: `https://github.com/snexus/nlp-question-answering-system`
4. Sentiment analysis model and evaluation: `https://www.kaggle.com/ananysharma/sentiment-analysis-using-bert`
5. `https://towardsdatascience.com/nlp-building-a-question-answering-model-ed0529a68c54`
6. `https://blog.paperspace.com/how-to-train-question-answering-machine-learning-models/`
7. `https://nlp.gluon.ai/master/tutorials/question_answering/question_answering.html`