

Catch Them If You Can

A Simulation Study on Malicious Behavior in a Cultural Heritage Question Answering System

A Master Thesis for the
MSc Business Analytics by:

Nikita GALINKIN

Supervised by:

Dr. Lora AROYO
Vrije Universiteit Amsterdam
Faculty of Sciences
De Boelelaan 1081a
1081 HV Amsterdam

Dr. Zoltán SZLÁVIK
IBM Nederland BV
Johan Huizingalaan 765
P.O. Box 9999
1066 VH Amsterdam

Abstract. Recent advances in artificial intelligence and machine learning have allowed question answering systems to become much more prominent for retrieving information to handle day-to-day tasks. This explorative study is done in the context of the collaborative SQALPEL project. In the study, we investigate the impact of malicious behavior in question answering systems specifically deployed in the cultural heritage domain. We review current practices and analyses from the literature and discuss how they apply to this study and their advantages and shortcomings. We find out that with over 30% of malicious users the system performance is influenced negatively. The impact of peaks of malicious activity has a negative effect that decreases over time. We propose filtering out malicious data as the most promising defense strategy.

1 Introduction

The desire for knowledge is hidden deep inside us, but our abilities to know and remember everything are limited. This is why almost immediately since the invention of the computer people started to develop automated systems that can understand questions and answer them in natural language, like a human would. The first systems were introduced in the early 1960's., one of them was a system called BASEBALL that answered questions about baseball games (Green et al., 1961).

Recent advances in artificial intelligence and machine learning, like IBM Watson¹ and Google DeepMind², have allowed question answering systems to become much more prominent for retrieving information to handle day-to-day tasks. Voice controlled assistants like Cortana³, Alexa⁴, Google Home⁵, Siri⁶, listen to people's commands or questions and perform the required actions to complete the task or provide the answer. Question answering systems integrated in messenger platforms, also called chatbots, perform a wide range of tasks from providing users with daily weather (e.g. Poncho chatbot⁷) or news updates (e.g. Washington Post bot⁸) to replacing customer service representatives in answering frequent inquiries (e.g. Lufthansa's bot named Mildred⁹).

In cultural heritage, question answering systems can serve as an entertaining and educating experience. However, the information provided by multimedia systems in cultural heritage is curated

¹ <https://www.ibm.com/watson/>

² <https://deepmind.com/>

³ <https://www.microsoft.com/cortana>

⁴ <https://developer.amazon.com/alexa>

⁵ <https://madeby.google.com/home/>

⁶ <https://www.apple.com/ios/siri/>

⁷ <https://poncho.is/>

⁸ <https://www.messenger.com/t/washingtonpost>

⁹ https://www.lufthansa.com/us/en/lufthansa_bot

by domain experts and often visitors may experience detachment, because they do not possess the same level of expertise and do not speak the same language. It was shown on a video labeling game that the overlap in vocabulary used by experts and lay audiences can be as little as 8% (Gligorov et al., 2011). Question answering systems can be used as a tool to explore the perspective of the audiences, gain their voice and through that improve the experience and learning outcomes.

As the example of Microsoft’s Twitter chatbot Tay teaches us, giving people a voice comes at the price of data reliability (Lee, 2016). Tay was built to chatter with Twitter users and learn from the conversations. However, learning from user interactions the chatbot quickly began to post offensive tweets forcing Microsoft to shut Tay down only 16 hours after the launch. We need to be prepared that some of the users will be malicious and will try to make the system unusable for other users by attacking it. We need to estimate the impact of their actions and study ways to deal with this problem. Recently the problem of malicious attacks has been broadly studied in the machine learning community (Barreno et al., 2008) (see section 3 for more detail). It has been shown that in some cases 20% of malicious data can lead to a ten-fold increase in classification errors (Xiao et al., 2015). The main trend in those studies is to investigate the worst-case scenario of the machine learning model with the assumption that the attacker has full knowledge of the system and his actions are optimal. These assumptions rarely hold in a system used in real-life. In this study, we propose a user model to simulate users attacking the system to gain a better indication of what might happen.

Our main research question for this work is:

RQ What is the impact of malicious feedback on the performance of a closed-domain, retrieval based question answering system?

This research question can be divided into sub-questions:

RQ1 To what extent does malicious feedback negatively influence system performance?

RQ2 What are the ways to influence this impact?

RQ1 will be answered through the experiments. **RQ2** will be answered based on the literature review and the findings of the experiments.

This explorative study is done in the context of the collaborative **SQALPEL** project. In the study, we investigate the impact of malicious behavior in question answering systems specifically deployed in the cultural heritage domain. We review current practices and analyses discussed in literature and discuss how they apply to this study and their advantages and shortcomings. We hypothesize that malicious behavior has a negative impact on question answering system performance, explore the severity and causes of this impact through a simulation and propose solutions backed by the state of the art approaches described in literature.

2 The SQALPEL project

As a use case to study the impact of malicious feedback on question answering systems we use the **SQALPEL** system, developed during the like named project. The **SQALPEL** project is a collaboration between IBM Center for Advanced Studies¹⁰ in Amsterdam, the Mauritshuis¹¹ museum in Den Haag and the Vrije Universiteit Amsterdam¹². The objective of the project is to make use of new techniques to answer and raise new questions about the subjects of a painting, relevant historical context and changing interpretations over the course of time in the art historical literature. The **SQALPEL** system uses the IBM Watson services Natural Language Classifier¹³ and Retrieve and Rank¹⁴ to retrieve the best answer to a user query (See Appendix B for diagrams explaining system

¹⁰ <https://researchweb.watson.ibm.com/university/cas/benelux/>

¹¹ <https://www.mauritshuis.nl/>

¹² <https://www.vu.nl/>

¹³ <https://www.ibm.com/watson/services/natural-language-classifier/>

¹⁴ <https://www.ibm.com/watson/services/retrieve-and-rank/>

architecture). This classifies the system as a retrieval based system (Clark, 2016), because we do not generate answers. The system can also be classified as a non-factoid, a system that produces not only facts as output, and closed domain, a system that answers questions only on a certain domain.



Fig. 1. Screenshot of the SQALPEL iOS application

The user can interact with the SQALPEL system through a chat interface implemented in an iOS app (see Fig. C3 for a screenshot). In the iOS app, the user has the options to input a question through voice or text, receive an output in voice or text and give binary feedback on the output in form of a like or dislike.

2.1 Use Case – SQALPEL system

During the project, we have developed a deep art history question answering system named SQALPEL, that is designed to answer questions about Rembrandt’s famous painting “The Anatomy Lesson of Dr. Nicolaes Tulp”¹⁵. The Anatomy Lesson of Dr. Nicolaes Tulp owned by the Mauritshuis was chosen for this project by the Mauritshuis as it has been widely studied in art historical context over the course of time. Much has been published about its commission, its provenance, who is represented, its conservation treatments, and changing art historical interpretations (see for instance (Afek et al., 2009)). As indicated by experts from the Mauritshuis, visitors

¹⁵ https://en.wikipedia.org/wiki/The_Anatomy_Lesson_of_Dr._Nicolaes_Tulp

often ask what medical procedure is depicted and how accurate the anatomy is, or they want to know more about the sitters and where else they might be depicted.

2.2 Data Set

Table 1. Description of the data set

	Total	By Source	
		Interview	Literature
Questions	3864	2246	1618
Answers	554	131	423
Mean number of Questions per Answer	6.98	17.15	3.83
QA Pairs	4037	2344	1693

In Table 1 the description of the data set is shown. The data for this study consists of 4037 unique question answer pairs with 3797 questions and 554 answers. The number of question answer pairs is higher than the number of questions because some different answers have the same question.

131 of the answers were collected from interviews with museum visitors, the rest we have extracted from literature about the painting (See Appendix B for a diagram explaining the extraction process). Answers from both sources were enriched with questions through the crowdsourcing platform called CrowdFlower¹⁶, where human workers were given the answer and had to come up with interesting questions that are answered by the given answer. A more detailed description of the data and the data gathering process can be found in Appendix A .

Questions collected from the interviews are more frequently asked, thus, on average, there are four times more questions to each interview answer. We will consider this fact when simulating the questions users ask.

3 Background about Malicious Attacks on Machine Learning Algorithms

The ultimate goal of artificial intelligence is to have a system that constantly adapts to its ever-changing real-life environment. Therefore, the system has to monitor its environment and learn from it. The threat of attacks on learning algorithms from outside intruders is a big concern in the machine learning society and the security of machine learning has been widely studied (Barreno et al., 2008). The field of machine learning security is called adversarial machine learning.

A taxonomy of attacks has been worked out in (Barreno et al., 2006). The problem of malicious user feedback in the SQALPEL system fits into this taxonomy in the following way:

- **Causative**, since the user actions have an influence on the training data
- **Indiscriminate**, because the user aims to make the system unusable. Targeted attacks on the SQALPEL system will result in just one question being answered poorly. This might be interesting for an attacker if he can inject his own malicious answer into the data set, which is not possible in the SQALPEL system.
- **Availability**, because the user aims to make the system unusable.

The earlier mentioned example of Microsoft Tay can be seen as a causative, indiscriminate, availability attack for the same reasons.

¹⁶ <https://crowdfLOWER.com/>

Causative attacks on different machine learning have been extensively discussed in literature. Works can be found about causative attacks on Machine Support Vector Machines (Biggio et al., 2011b, 2012; Laishram and Phoha, 2016; Burkard and Lagesse, 2017), Deep Neural Networks (Papernot et al., 2017), Clustering algorithms (Biggio et al., 2013), Logistic Regression and Linear Regression (Mei and Zhu, 2015). Other parts of the machine learning pipeline have also been studied for possible attacks, like, for example, the robustness of feature selection algorithms against training data poisoning (Xiao et al., 2015). In these works the authors look at different attack strategies and test their proposed defense strategies.

Most of this research on attacks, however, assumes that the attacker has full knowledge of the model he attacks (Šrndić and Laskov, 2014) and that his actions are optimal. In other words, worst-case scenarios are investigated that are unlikely to happen in reality. Our aim is to provide an insightful study to potential users of question answering systems in the cultural heritage domain, thus our assumptions are focused on more realistic settings. We assume that the attacker has little knowledge about the model he attacks and his actions are not optimal.

Research in this domain is not only exploratory, but also tries to propose proactive or reactive defense mechanisms against the discussed attacks. In their work (Barreno et al., 2006) discuss defense strategies for each possible attack classified by their taxonomy. The proposed proactive defenses, however, are based on the idea of adapting the algorithms to be prepared for malicious actions, like regularization or randomization, something we have no possibility to do, since the algorithms in the SQALPEL system are treated as a black box.

Another proactive strategy is the use of multiple classifiers, called bagging. Bagging is known to be more robust against outliers and is shown to be a very promising defense strategy against causative attacks (Biggio et al., 2011a).

We should also mention a large area of research on the proactive side called adversarial learning. Here the approach is to include adversarial examples into the training set to make the algorithm more robust against potential attacks (Kurakin et al., 2016; Goodfellow et al., 2014; Huang et al., 2015). For our use case this is not possible, because the attacks we are investigating do not insert new data into the training set, but rather gain an impact through the feedback they give.

Reactive defense strategies require the detection of an attack. For causative attacks, an attack classifier can be trained on a special test set that includes known intrusion variants. The best strategy of the learner in this case is to ignore the detected malicious points (Barreno et al., 2006). Another strategy is to flag parts of the training data likely to be attacked and focus human expert attention on those parts (Mei and Zhu, 2015). We will discuss both approaches as they are most promising in our study.

To our knowledge no work has been previously done about malicious attacks on question answering systems.

4 Experimental Design

We investigate the impact of malicious feedback on question answering systems by simulating system users and evaluating the data resulting from this simulation.

A simulation is used because at the time of the study the iOS application was not in a stage to be publicly displayed making a study with real users not possible.

To run the simulation we need to follow these steps:

- **Split the data** into training, simulation and validation set
- Decide on the **settings** of the experiment, e.g. its duration, based on project requirements
- Simulate users interacting with the SQALPEL system based on a **user model**
- **Evaluate system performance** to measure the impact of malicious feedback

See Appendix D for a link to the implementation of the simulation.

Further we present a detailed overview on each of these steps.

4.1 Data

The user interaction simulation needs a well-trained initial system, similar to the initial system that will be presented at the museum. Users will be able to interact with this system by asking questions, receiving answers and giving feedback. We divide our data into three data sets to simulate this behavior:

- The **initial training set** contains question answer pairs that are used for training the initial system that visitors will see in the museum on day 1.
- The **simulation set** contains question answer pairs the system has not seen during initial training.
- The **validation set** contains question answer pairs that the system has not seen during our simulation.

We split the data with a target ratio of 60/20/20:

Table 2. Size of data set in number of unique questions; percentage of answers covered by questions in data set; distribution between sources of questions in set

Set	Size	% of total	Answer Coverage			Source Distribution			
			Total	Int.	Lit.	Int.	% in set	Lit.	% in set
initial	2399	62.1%	99.7%	99.2%	99.8%	1226	51.1%	1173	48.9%
sim	884	22.9%	80.3%	97.7%	74.9%	543	61.4%	341	39.6%
val	581	15.0%	40.6%	98.5%	22.8%	477	82.1%	104	17.9%

The distribution of question type (what the question is asking for – a description, a location, a person etc.) and the distribution of number of questions per answer are kept as close as possible in each data set. This is the reason why the resulting ratio is not exactly corresponding to the target ratio.

The three sets are distinct on the question side to make sure that the questions are not shown to the system in the wrong phase of the simulation. The answer coverage is most important in the initial training set, because the system will not be able to come up with the right answer if it does not know it. In the other two sets the answer coverage is important to make sure our results are not biased towards a certain set of answers.

4.2 Experimental Settings

The SQALPEL system will be displayed in the Mauritshuis on two iPads in front of the painting. The museum is open 7 days a week for 8 hours. In this time, daily on average 200 users can spend 5 minutes each with one of the iPads. We simulate two weeks of operations with 200 daily users, each user simulated according to the user model. The duration of two weeks is chosen pragmatically due to the limited time of the project. If no trend is visible in that period we continued the simulation for a longer time to be able to draw conclusions.

In Fig. 2 a visualization of the simulation process is shown. The simulation starts with a QA system that is initially trained and can be explained in a number of steps:

1. **Simulate a user profile** consisting of the user’s intent, the number of questions the user asks and his fallibility.

2. **Simulate user’s questions** by randomly drawing them from the initial training set and the simulation set with equal probability and without replacement, since one user is unlikely to ask the same question twice. Both sets are used because users may ask questions that the system is initially trained on or questions that have never been seen before.
3. **Retrieve an answer** to each of them by querying the SQALPEL system. We sample 80% of the questions from the interview data set and the rest from the literature data set, based on an estimation given by the museum’s guides.
4. **Simulate feedback** based on the intent and the fallibility of the user. If the user is malicious he will give feedback opposite to the ground truth with a probability of 82% (his fallibility). If he is not malicious he will give feedback in agreement to the ground truth with the same fallibility. The feedback is stored in the systems database.
5. **Simulate next user.** This loop is repeated 200 times simulating 200 different users. The feedback of all 200 users together makes up one day of operation in the museum.
6. **Retrain SQALPEL system.**
7. **Evaluate performance** by querying system with each question from the simulation and validation sets and store each answer in a results file. With the evaluation of the retrained system the epoch consisting of one day of system operation at the museum is completed.
8. **Simulate next day.** This loop continues until all days are completed.

4.3 User Modeling

User modeling is a well-studied topic frequently used in the areas of Information Retrieval and Question Answering.

In Question Answering statistical user models are used for dialogue management with the aim of enriching the training set with simulated dialogue (Schatzmann et al., 2006).

In Information Retrieval multi-dimensional user models are used to simulate user behavior to draw conclusions on the system-user interaction (Keskustalo et al., 2006; Baskaya et al., 2011). Multi-dimensional models have also been applied in question answering studies with the same goal (Quarteroni and Man, 2006).

The aim of our simulation is similar to the one in the works of (Keskustalo et al., 2006; Baskaya et al., 2011), thus we use a three-dimensional user model inspired by these works, with the following dimensions:

1. **Intent.** A binary dimension that describes whether a user is malicious or not. We define malicious feedback as feedback on a question answer pair that is opposite from our ground truth. A malicious user is a user who gives malicious feedback. *User intent is modeled by a binomial distribution with probability equal to the fraction of malicious users.* This is the parameter we will vary over the different experiments to see what the impact of malicious behavior is.
2. **Amount of questions asked.** It might be hard to come up with questions about the painting and some users will only ask one or two general questions while other users will ask more. Considering the project goal of a 5-minute user interaction, we can calculate how many questions on average a user can type in and read in this time. The average question length in our data set is 9 questions and the average human typing speed on a large touch screen is 32 words per minute (Sears et al., 1993). The average answer length in our data set is 43 words and the average reading speed of a human is 250 words per minute (Bell, 2001). Thus, in approximately 5 minutes a museum visitor can type and read 10 questions and *we model the distribution of the amount of questions asked by a normal distribution with mean 10 and standard deviation 2.*

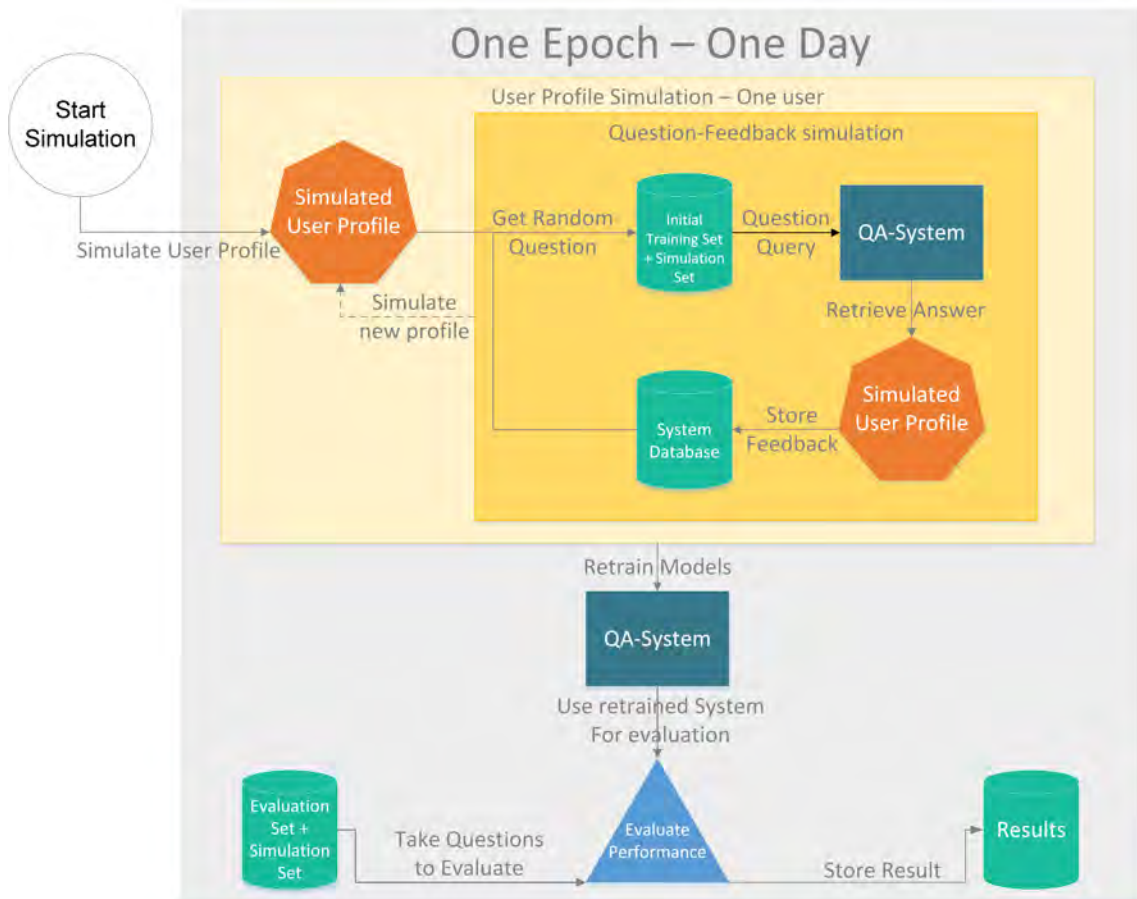


Fig. 2. Visualization of the simulation process

3. **Fallibility.** Since humans are not consistent in their judgment (Baskaya et al., 2011) we need to model errors in the feedback users give. On a document relevance assessment task similar to ours, the agreement between human workers and expert judges has been found to be 0.82 for the top ranked document (Al-Maskari et al., 2008). *We use this finding to model human fallibility through a binomial distribution with probability 0.82.*

We simulate only users who give feedback, since the users who do not give feedback do not influence the system.

4.4 Evaluation

Multiple metrics have been used in literature to evaluate performance of retrieval based systems. The metrics used range from well-established approaches like Mean Average Precision at rank N (MAP@N) together with Recall or more exotic approaches that take into account how much a user has to work to get to the correct result (Radev et al., 2002). Most of the metrics, however, assume that multiple answers are displayed to the user which is not the case with the SQALPEL system. We show the user only the top matched result as the answer to his question. Hence, MAP@1 or simply MAP is the metric we use to understand the impact of malicious behavior.

MAP measures the percentage of questions that were answered correctly. It has been shown that precision is highly correlated with user satisfaction (Al-Maskari et al., 2007). Unfortunately, there have been no studies on user tolerance towards the precision of a retrieval based system. Thus we can not exactly say at which point the system becomes unusable to a user.

Measuring MAP requires us to know exactly which answer is correct to which question. This is a problem for us because the data set used in this study is lacking labels, only a fraction of all possible correct answers to each question are annotated. To overcome this problem, we have built a classifier that can classify an answer as correct or not with a high precision.

The classifier we built was inspired by the question answering evaluation metric POURPRE that measures the co-occurrence of words between a given answer and an answer nugget. An answer nugget is defined as a string containing a fact for which the assessor could make a binary decision as to whether a response contained that nugget (Lin and Demner-Fushman, 2006). In the crowdsourcing task aiming to collect questions, users were asked to highlight the part of the answer that answers their question. These annotations we use as the answer nugget.

More details on the answer correctness classification can be found in Appendix E .

5 Results and Discussion

In this section we discuss the results of the experiments answering **RQ1** and propose ways of dealing with it answering **RQ2**.

5.1 Experiment Results

We have run a simulation with different levels of maliciousness to see what impact it has on the system and how big that impact is.

Overall Results In Fig. 3 we can see the MAP of the system for the different levels of maliciousness. The different graphs show the MAP on the validation set and simulation set together and separately.

We can see that with up to **10% of malicious users** system performance is improving over time. This means that a system with 10% of maliciousness is not in danger of becoming worse. The performance increase seems to be almost linear. We expect it to slowly converge towards the

maximum performance of the system. Further experiments are needed to determine what the maximum performance is.

With **20% - 30% of malicious users** system performance appears to stagnate. Here we have extended our simulation period to 26 days and can see that the trend is slightly positive overall, but slightly negative on the validation set.

With a higher number of malicious feedback system performance decreases rapidly. In one day the performance loss observed with **over 30% of malicious users** is higher than the gain without malicious users over the whole period of all 14 days. This means that if there is a large amount of malicious data on day one of system usage the damages will be such, that for more than two weeks the system performance will stay below the performance of the initial system. The performance is decreasing very suddenly at the beginning and continues to decrease at slower pace as time progresses. We expect a system with over 50% of malicious users to converge towards a precision of 0% as the majority of the feedback is malicious and it will take over the good feedback eventually. With 30% to 50% we expect the performance to go down to a certain level above 0 and even out. System performance will not return to normality, since a question answer pair with more negative than positive feedback will be ignored at retraining and will have no chance of finding its way back to the training data.

On the simulation and on the validation set the performance of every system is consistently different with, surprisingly, the performance of the validation set being higher. This can be explained by the answer coverage of the validation set – it is biased towards the interview data, the part of the system that performs better. Overall on both sets we see very similar behavior, with steeper learning curves on the simulation set. This is logical, since the system has seen those questions and if the feedback was good it has learned to answer them.

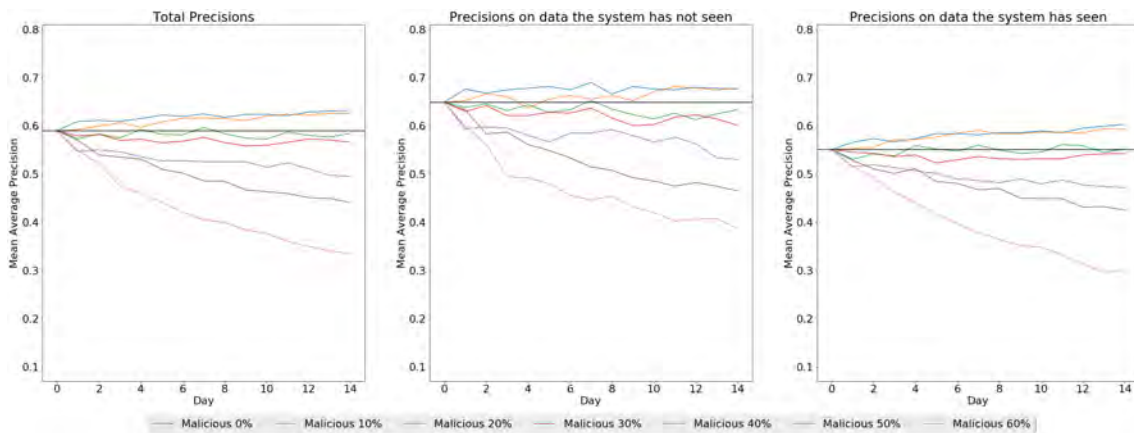


Fig. 3. MAP of the system measured after each day of the simulation;

1. Precision on simulation + validation set;
2. Precision only on validation set;
3. Precision only on simulation set

The black line represents the performance of the initial system

Interview Results In Fig. 4 we can see the performances on questions from the interview data. We see here an overall improvement for up to 30% of malicious feedback even over the course of 14 days. For 20% - 30% malicious users the system learns well on the seen data, but performance decreases slightly on the unseen data. This is an expected behavior, since the model overfits the parts of the seen data that have been randomly drawn during the simulation. Besides this we observe the same behavior as on the system overall.

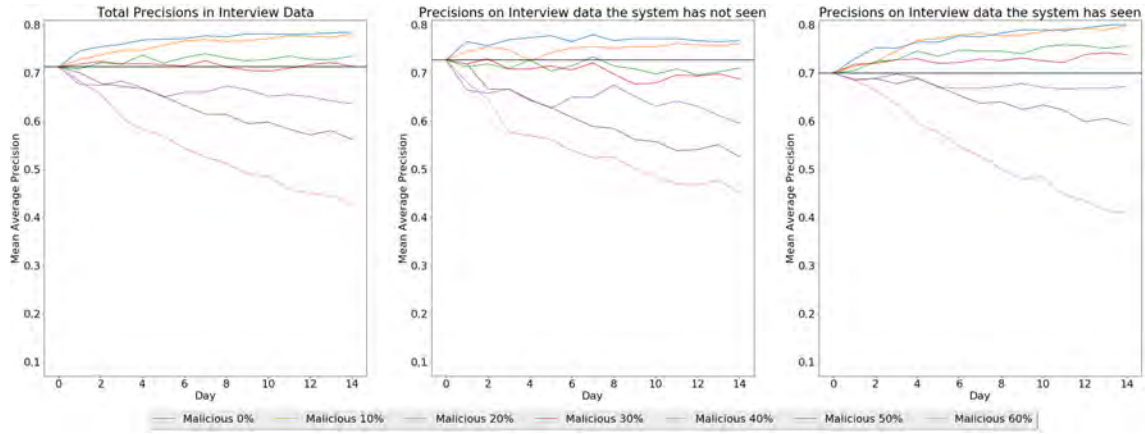


Fig. 4. MAP of the system (only Interview data) measured after each day of the simulation; 1. Precision on simulation + validation set; 2. Precision only on validation set; 3. Precision only on simulation set
The black line represents the performance of the initial system

Literature Results In Fig. 5 we see the performance of the system on the data from the literature. Interestingly, we see a decrease in performance for every experiment. There are a number of reasons for that. First of all, the data set is very large and in the beginning every feedback received will have a large impact on the performance. Secondly, questions from this source are asked rarely, resulting in an even smaller amount of feedback. This leads to a learning behavior that is much slower than that of the interview data. Due to these qualities this part of the system becomes more vulnerable towards malicious attacks, but at the same time the questions in this data set are more in-depth. Coming up with questions targetting this data set requires more effort, something a malicious user with the aim of quickly destroying the system is unlikely to do.

On the unseen data we notice a very chaotic behavior for experiments with up to 40% of malicious users. The reason for this is again that the validation set covers only a small percentage of the literature data.

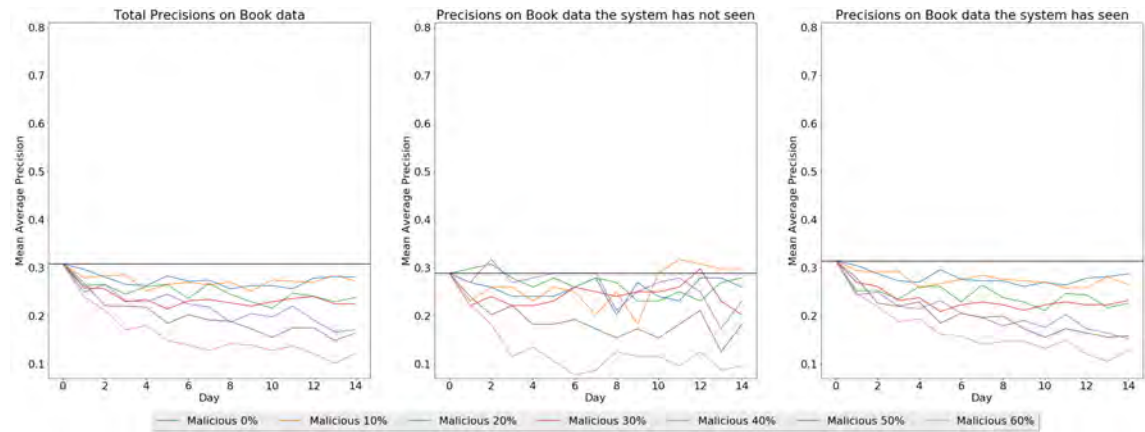


Fig. 5. MAP of the system (only literature data) measured after each day of the simulation; 1. Precision on simulation + validation set; 2. Precision only on validation set; 3. Precision only on simulation set
The black line represents the performance of the initial system

Peak Results In the SQUALPEL project the initial aim is to make the question answering system available solely on two devices in front of the painting. A more realistic scenario in this case is a constantly low percentage of malicious users with a possible peak because a certain group of people with a low quality pays a visit to the museum during one of the days. To see what happens in such a scenario we have conducted experiments where the amount of malicious users is constantly at 10% with a peak of 60% on one day. We have chosen a low and high percentage from the previous experiment to be able to relate to previous results. We have performed two different experiments, one with a peak at day 3 and one with a peak at day 12. This difference is made to see whether a peak close to system launch has a bigger effect because at that time the system has received little feedback and the impact of each malicious user is higher. Each of the two experiments has been performed three times, results are averaged over each day.

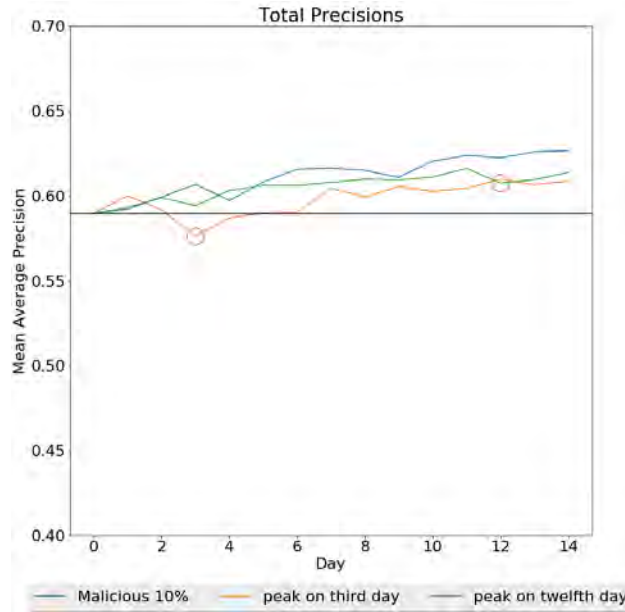


Fig. 6. MAP on the simulation and validation sets for peaks of malicious attacks

In Fig. 6 the system performance with the peaking malicious days is shown together with the performance of the experiment without a peak, for comparison. We can see that no matter when the peak occurs in both cases the performance ends up at the same level, however in the time between the two peaks, day 3 until day 12, the system where the peak was earlier is constantly performing poorer. Both systems end up performing worse than the system that does not see this peak.

Table 3 shows the p-values of the Kolmogorov-Smirnov test that we did to see whether the results of the experiments are significantly different. The null hypothesis of this test is that the two samples come from the same distribution. With a low p-value the null hypothesis is rejected indicating a potential difference in distributions (marked by the bold values in the table).

The results of the tests show the same behavior as seen in Fig. 6. Before the peak the performance of the peaked experiments is not significantly different from the performance of the experiment without the peak. Between day 3 and day 12 the experiment with the early peak shows a different performance from the other two. After day 12 the experiments with the peak show a performance that is not significantly different from each other, but different from the experiment without the peak.

We expect the systems with the peak to catch up only after the system without the peak flattens out. It is, however, unclear whether they will flatten out at the same performance level, as we do not know whether the correct question answer pairs that have been excluded due to bad feedback

Table 3. p-values of the Kolmogorov-Smirnov test used to determine similarity between two samples

First Sample	Second Sample	Days			
		0 - 2	0 - 11	12-14	0 - 14
10% Malicious	Peak Day 3	0.976	0.005	0.011	0.005
10% Malicious	Peak Day 12	0.976	0.433	0.011	0.0511
Peak Day 3	Peak Day 12	0.976	0.019	0.976	0.051

in the peak days will be relearned eventually. To understand this further runs and investigations are required.

Another interesting observation from the experiments with the peaks is that the impact of the 60% of malicious users decreases the later they appear. While at day 0 they cause a 7% drop in performance of the previous day, at day 3 the drop is 2.7% and at day 12 it is only 1.4%. This indicates that for such an attack only in the first few weeks system security has to be monitored closely and active intervention is necessary in case misbehavior is detected. After that initial period a sudden peak will not harm the system significantly and can be ignored. This happens due to the large amounts of non-malicious feedback already present in the system. This non-malicious feedback does not allow the malicious feedback to trigger significant changes. A constantly high level of maliciousness, however, needs to be dealt with, because if the effect persists long enough the malicious feedback will start dominating the non-malicious feedback.

5.2 Defense Strategies

In the literature, we have seen several proactive and reactive defense strategies that aim to eliminate the effect of malicious attacks. Based on (Barreno et al., 2006) the most promising reactive strategy against such causative attacks is to identify the malicious users and exclude them from the data.

A very radical reactive approach is to delete all the feedback received in one day of the presence of a large number of malicious users. This will delete all good and bad feedback and will bring the system to the performance of the previous day. This approach will be successful in the case of peaking days of maliciousness, as the negative effect observed in the corresponding experiments will be eliminated. In case of a constantly large number of malicious users this approach will lead us to a static system that will be the same as the initial system.

A more granular way to handle the data is to try and identify the users that are malicious and filter only the feedback given by them, leaving all good feedback in the system. Possible approaches here include the score or the majority vote. However, these techniques will overlook the possibility that some of the answers are difficult to judge and users performing bad on those answers will be punished. Therefore using these approaches will result in filtering out more users than really are malicious which will slow down the learning process.

We think that the task of giving feedback like the one used in the SQALPEL system is comparable to a crowd sourcing task where human workers annotate data for training and evaluating of computational systems. Thus, an approach like the CrowdTruth approach proposed in (Aroyo and Welty, 2014) can be helpful in identifying users who give low quality feedback. The novelty behind CrowdTruth is to evaluate quality not solely based on annotator agreement, but taking into account the difficulty of each task and each relation to be annotated.

A downside of filtering overall is that it relies on the assumption that the bigger fraction of the crowd behaves well. This means that if the fraction of malicious users is larger than 50% filtering algorithms will think that the feedback that the malicious users give is correct and will start filtering out the well-behaved users.

A third reactive approach, as proposed by (Mei and Zhu, 2015), can be to flag potentially malicious users (e.g. again through CrowdTruth) and leave the final decision to experts rather than to make it automatically. Here a team of experts would daily receive the flagged question answer pairs for review and will have to decide manually on the correctness of the answer given the question. The workload with this approach will quickly increase and it is safe to say that in the domain of cultural heritage this approach is infeasible.

As seen in the literature proactive defense strategies are mostly on algorithm level and thus are not applicable in our case.

Introducing user accounts can also be seen as a proactive defense mechanism. The intuition behind this is that without user accounts users are tracked solely by their session id. This means that with opening up a new session they are seen as a new user, thus, if in each session the workload they complete is low, their cumulative effect on the system will be the same as if there was one user with a high workload. But users with a small workload are more difficult to detect for a filtering algorithm and thus their success rate might be higher. With user accounts the motivation of creating many users can be lower and thus this serves as a defense mechanism. However, the system will lose its anonymity and this might lead to users thinking more about what questions they ask. This will have a restrictive effect on the goal of giving people a voice. Also introducing a registration procedure on-site will have a negative effect on visitor willingness to use the system.

Another downside can arise if user accounts are introduced with the goal of flagging spam users. Whenever a spammer is flagged the system will prevent him from giving more feedback. This serves as motivation for the user to create a new account. Thus, the result of our prevention mechanism is the rising number of malicious users. As our experiments have shown, this will lead to a lower system performance and will fail to protect the system.

6 Conclusions and Future Work

In this work we have used a simulation on the SQALPEL system to explore the impact of malicious feedback on a question answering system especially developed for the cultural heritage domain. The motivation for this study comes from the recent example of Microsoft Tay, a chatbot that was made unusable within less than 24 hours. Following this example our study focused on the initial stage of the system when it is empty and most vulnerable.

We have formulated two research questions:

- RQ1** To what extent does malicious feedback negatively influence system performance?
- RQ2** What are the ways to influence this impact?

An extensive answer to **RQ1** can be found in section 5.2 and a discussion on **RQ2** can be found in section 5.1.

We have shown that with up until 20% of malicious feedback the system continues to learn and system performance does not drop below the level of the initial system. With over 30% of the users being malicious system performance will gradually decrease. With 60% of malicious feedback it drops by a factor of 2 after only two weeks.

We have looked at a scenario where a peak of low quality feedback appears and have investigated whether the timing of the peak makes a difference. The results indicate that a peak in malicious activity negatively influences system performance and creates a long term effect. The timing of the peak defines its influence on system performance. The later the peak occurs the less effect it will have.

To protect a question answering system from malicious feedback the most promising strategy appears to be filtering. Hereby multiple ways to filter out malicious feedback are possible with

a different level of granularity. In general the rule is - the less data we filter out, the better our performance will be after filtering. Further study is necessary to compare the effect of the proposed approaches.

For somebody who plans to deploy a question answering system without having the needed technical staff in house the results of this study show that in the first weeks of system operation user behavior and misbehavior have to be watched closely as the potentially negative effect on overall system performance can be severe.

For future work we propose to study the problem with real users. Two different experiments are possible and both can help us develop a better understanding of the problem.

A study much like this one but with real users in a controlled environment can help in understanding whether there might be ways to attack the system that we have not considered. Here users can be divided into two groups, based on their intent. Malicious users will be told to try and make the system unusable any way they can imagine and good users will be told to operate in good faith.

A study on site in the museum can help understand how the users interact with the system in reality. From that we can learn the real percentage of malicious users, whether the assumption of peaks actually happens, how long users interact with the system, what the users willingness to provide feedback is and what their fallibility is.

In both studies the proposed defense mechanisms can be tested and their impact can be compared.

It is also important to understand the dynamics of the impact of malicious behavior through time. Possibly the system becomes more robust against malicious feedback the more feedback is saved in the database.

Our study is aimed only at attacks through malicious user feedback. But other potential attacks should be also considered and explored. For example users could try to ask irrelevant questions that the system will classify as a question about the painting and might even retrieve an answer to that question with a high confidence. If this question answer pair then receives positive feedback it lands in our training data. With an optimal strategy a user could potentially erase all good questions and replace them with his irrelevant questions also pushing the system to become unusable.

Another interesting study is aimed at developing an understanding of the performance threshold at which the system is perceived as unusable by potential system users. It is, however, questionable how long the results of such a study will be valid. The intuition tells us that users will get more demanding as AI progress continues and the average performance of question answering systems rises.

7 Acknowledgments

I would like to thank Zoltán Szlávik for keeping me organized, managing my expectations and for always being there. Thank you Lora Aroyo for your passion, your expertise, your wisdom and for always pushing me forward. Dorottya Mezőfi thank you for the talks we had, thank you for helping with everything and thank you for always supporting me. Thank you to Bob de Vries who was my partner in crime on the SQALPEL system. Thank you to Hedwig Wösten, Boudewijn Koopmans and Charlotte Wytéma for your endless curiosity and excitement. And for the great museum tour! A big thank you to the people from CAS: Manfred Overmeen, Benjamin Timmermans, Madli Uutma, Nikolay Polyanov, Gregory Afentoulidis, Maximilian Lombardo, Davide Aurucci, Zsuzsanna Szabo and Jordy Alblas. Last but not least a big thank you to my family for all the love and the best support I could imagine.

Bibliography

- Afek, A., Friedman, T., Kugel, C., Barshack, I., and Lurie, D. J. (2009). Dr. tulp’s anatomy lesson by rembrandt: the third day hypothesis.
- Al-Maskari, A., Sanderson, M., and Clough, P. (2007). The relationship between ir effectiveness measures and user satisfaction. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR ’07, pages 773–774, New York, NY, USA. ACM.
- Al-Maskari, A., Sanderson, M., and Clough, P. (2008). Relevance judgments between trec and non-trec assessors. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR ’08, pages 683–684, New York, NY, USA. ACM.
- Aroyo, L. and Welty, C. (2014). The three sides of crowdtruth. *Journal of Human Computation*, 1:31–34.
- Barreno, M., Bartlett, P. L., Chi, F. J., Joseph, A. D., Nelson, B., Rubinstein, B. I., Saini, U., and Tygar, J. D. (2008). Open problems in the security of learning. In *Proceedings of the 1st ACM Workshop on Workshop on AISEc*, AISEc ’08, pages 19–26, New York, NY, USA. ACM.
- Barreno, M., Nelson, B., Sears, R., Joseph, A. D., and Tygar, J. D. (2006). Can machine learning be secure? In *Proceedings of the 2006 ACM Symposium on Information, Computer and Communications Security*, ASIACCS ’06, pages 16–25, New York, NY, USA. ACM.
- Baskaya, F., Keskustalo, H., and Järvelin, K. (2011). Simulating simple and fallible relevance feedback. In *Proceedings of the 33rd European Conference on Advances in Information Retrieval*, ECIR’11, pages 593–604, Berlin, Heidelberg. Springer-Verlag.
- Bell, T. (2001). Extensive reading: Speed and comprehension. In *The Reading Matrix*, 1(1), page 1–13.
- Biggio, B., Corona, I., Fumera, G., Giacinto, G., and Roli, F. (2011a). Bagging classifiers for fighting poisoning attacks in adversarial classification tasks. In Sansone, C., Kittler, J., and Roli, F., editors, *MCS*, volume 6713 of *Lecture Notes in Computer Science*, pages 350–359. Springer.
- Biggio, B., Nelson, B., and Laskov, P. (2011b). Support vector machines under adversarial label noise. In Hsu, C.-N. and Lee, W. S., editors, *Proceedings of the Asian Conference on Machine Learning*, volume 20 of *Proceedings of Machine Learning Research*, pages 97–112, South Garden Hotels and Resorts, Taoyuan, Taiwan. PMLR.
- Biggio, B., Nelson, B., and Laskov, P. (2012). Poisoning attacks against support vector machines. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*.
- Biggio, B., Pillai, I., Rota Bulò, S., Ariu, D., Pelillo, M., and Roli, F. (2013). Is data clustering in adversarial settings secure? In *Proceedings of the 2013 ACM Workshop on Artificial Intelligence and Security*, AISEc ’13, pages 87–98, New York, NY, USA. ACM.
- Burkard, C. and Lagesse, B. (2017). Analysis of causative attacks against svms learning from data streams. In *Proceedings of the 3rd ACM on International Workshop on Security And Privacy Analytics*, IWSPA ’17, pages 31–36, New York, NY, USA. ACM.
- Clark, M. (2016). Digital engagement blog. Retrieved July 18, 2017, from <http://info.contactsolutions.com/digital-engagement-blog/a-chatbot-framework>.
- Gligorov, R., Hildebrand, M., van Ossenbruggen, J., Schreiber, G., and Aroyo, L. (2011). On the role of user-generated metadata in audio visual collections. In *Proceedings of the Sixth International Conference on Knowledge Capture*, K-CAP ’11, pages 145–152, New York, NY, USA. ACM.
- Goodfellow, I. J., Shlens, J., and Szegedy, C. (2014). Explaining and harnessing adversarial examples. cite arxiv:1412.6572.
- Green, Jr., B. F., Wolf, A. K., Chomsky, C., and Laughery, K. (1961). Baseball: An automatic question-answerer. In *Papers Presented at the May 9-11, 1961, Western Joint IRE-AIEE-ACM Computer Conference*, IRE-AIEE-ACM ’61 (Western), pages 219–224, New York, NY, USA. ACM.

- Huang, R., Xu, B., Schuurmans, D., and Szepesvári, C. (2015). Learning with a strong adversary. *CoRR*, abs/1511.03034.
- Keskustalo, H., Järvelin, K., and Pirkola, A. (2006). The effects of relevance feedback quality and quantity in interactive relevance feedback: A simulation based on user modeling. In *Advances in Information Retrieval, 28th European Conference on IR Research, ECIR 2006, London, UK, April 10-12, 2006, Proceedings*, pages 191–204.
- Kurakin, A., Goodfellow, I. J., and Bengio, S. (2016). Adversarial machine learning at scale. *CoRR*, abs/1611.01236.
- Laishram, R. and Phoha, V. V. (2016). Curie: A method for protecting svm classifier from poisoning attack. *CoRR*, abs/1606.01584.
- Lee, P. (2016). Learning from tay’s introduction. Retrieved August 22, 2017, from <https://blogs.microsoft.com/blog/2016/03/25/learning-tays-introduction>.
- Lin, J. J. and Demner-Fushman, D. (2006). Methods for automatically evaluating answers to complex questions. *Inf. Retr.*, 9(5):565–587.
- Mei, S. and Zhu, X. (2015). Using machine teaching to identify optimal training-set attacks on machine learners. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, AAAI’15*, pages 2871–2877. AAAI Press.
- Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., and Swami, A. (2017). Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, ASIA CCS ’17*, pages 506–519, New York, NY, USA. ACM.
- Quarteroni, S. and Man, S. (2006). User modelling for adaptive question answering and information retrieval. In *In Proceedings of FLAIRS’06*.
- Radev, D. R., Qi, H., Wu, H., and Fan, W. (2002). Evaluating web-based question answering systems. In *In Proc. of LREC 2002*.
- Schatzmann, J., Weilhammer, K., Stuttle, M., and Young, S. (2006). A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies. *Knowl. Eng. Rev.*, 21(2):97–126.
- Sears, A., Revis, D., Swatski, J., Crittenden, R., and Shneiderman, B. (1993). Investigating touch-screen typing: The effect of keyboard size on typing speed. In *Behaviour Information Technology 12(1)*, pages 17–22.
- Šrndić, N. and Laskov, P. (2014). Practical evasion of a learning-based classifier: A case study. In *Proceedings of the 2014 IEEE Symposium on Security and Privacy, SP ’14*, pages 197–211, Washington, DC, USA. IEEE Computer Society.
- Xiao, H., Biggio, B., Brown, G., Fumera, G., Eckert, C., and Roli, F. (2015). Is feature selection secure against training data poisoning? In *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37, ICML’15*, pages 1689–1698. JMLR.org.

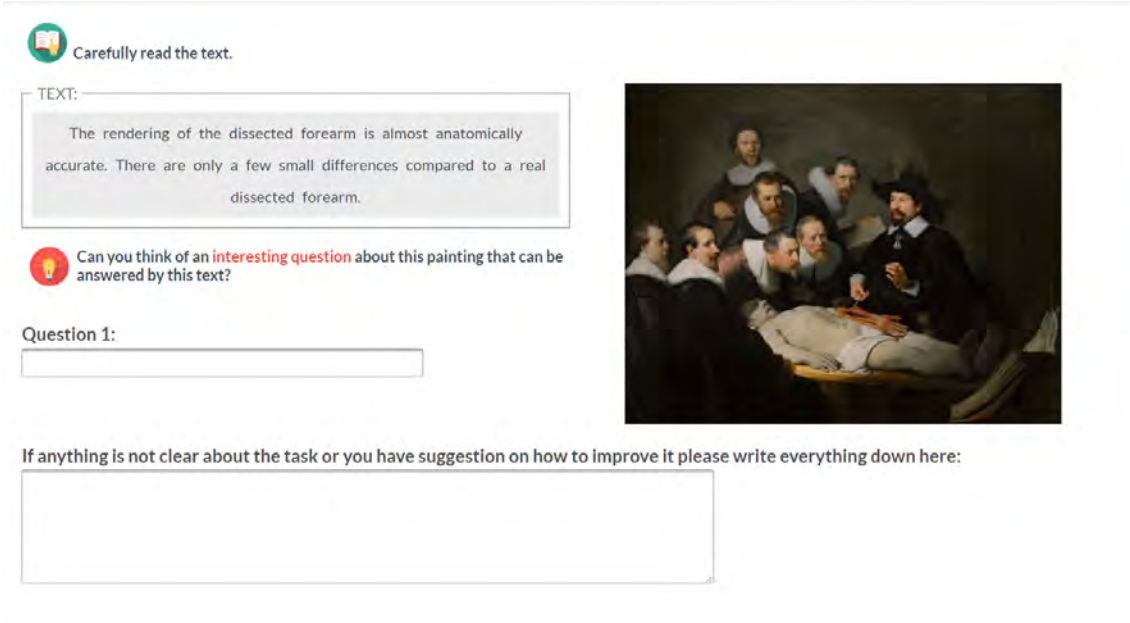
Appendix A: About The Data Used In The SQALPEL System

Data Gathering

At the start of the project we were given two audio files with interview recordings made during the museum night 2016 and 25 books in .pdf format.

We have transcribed and translated the interviews which resulted in 294 question-answer pairs. Those were given back to the Mauritshuis to fact-check and correct grammar. After the Mauritshuis confirmed an answer we made a crowdsourcing task around it. A screenshot of the task can be seen in Fig. A1. People were asked to come up with an interesting question about the painting that can be answered by the displayed answer. Additionally, people were asked to highlight in the answer field which words provide an answer to their question.

From the book data 11 books were selected. The books that were not selected were either not in English or there was no possibility to extract the text from the .pdf. Within the selected books some contained information not relevant for this painting (mostly about other paintings) – this information was discarded as it would have confused the system and could have lowered output quality. The selected pages were split into paragraphs, this resulted in over 4000 paragraphs. Some of the paragraphs contained no information, to exclude them a data set was manually created and a natural language classifier was trained to distinguish paragraphs that contain information and paragraphs that do not. After filtering we were left with 967 paragraphs. Those paragraphs were also run through the CrowdFlower task to collect questions to them.



Carefully read the text.

TEXT:

The rendering of the dissected forearm is almost anatomically accurate. There are only a few small differences compared to a real dissected forearm.

Can you think of an interesting question about this painting that can be answered by this text?

Question 1:

If anything is not clear about the task or you have suggestion on how to improve it please write everything down here:

Fig. A1. Screenshot of the task in CrowdFlower

Interview Data

As of July 9th the Museum has fact checked 153 answers and 133 of them have gone through CrowdFlower. In total this resulted in 2390 new questions, on average we have gathered 18.7 questions per answer. The quality of the questions is good, every worker did their best to come up with questions that are relevant. The quality of the answer selections is not so good, but at this moment it was not used anywhere in the project. In Fig. A2 a histogram of number of questions available per answer is shown. We can clearly see the line between the answers that have been

through CrowdFlower and the answers that have not. In the research only answers with more than 5 questions will be used.

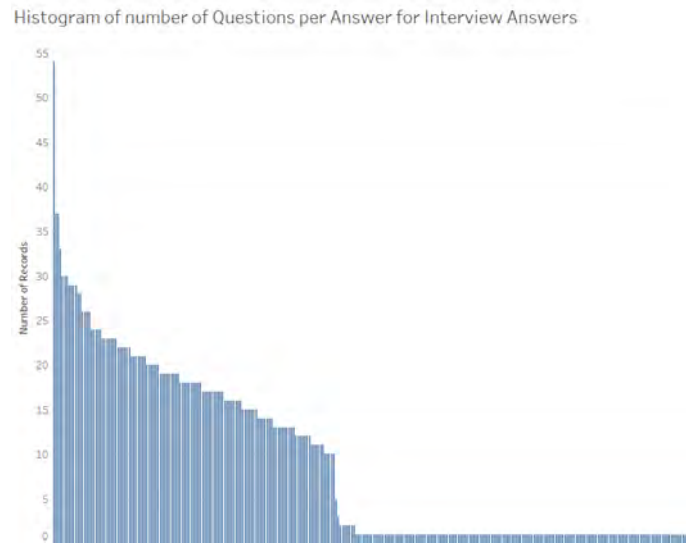


Fig. A2. Histogram of number of questions per answer for the interview data set

In Fig. A3 a detailed view on the answers that have 5 or more questions is provided. The colors distinguish the type of question, so what they are asking for (description, entity, person, location, numerical value). We can see that answers that have 10 questions or less usually have a clear question type associated, those are the shorter answers that contain mostly one fact and it is not possible to ask about anything else, here the questions are reformulations of each other. Answers with more questions have more question types associated, since they are longer and people can ask more varied questions. For example the answer with the most questions is about the book in the painting and contains information about the writer of the book, the year it was published, a description of the book, the location of the book in the painting, thus questions of each type are possible.

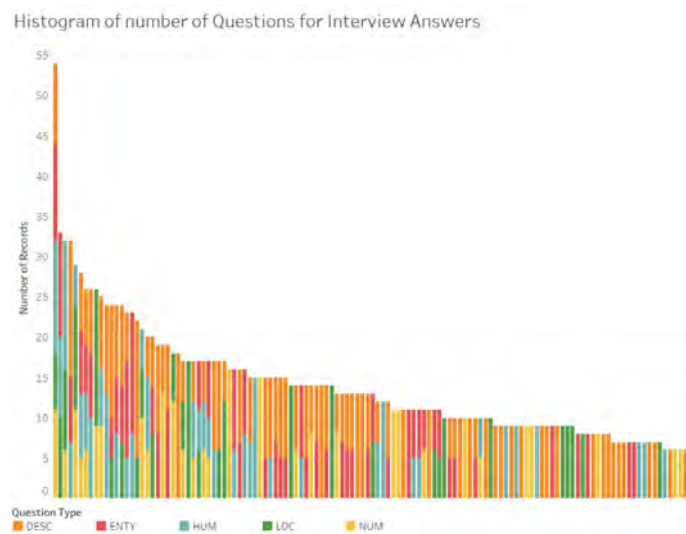


Fig. A3. Histogram of number of questions per answer for the interview data set including question topics

Literature Data

As of July 9th we have gathered 2495 questions for the book answers. The quality of the questions is good again, although poorer than in the interview case. Since answers extracted from books are mostly longer and can be incomplete, it is harder to pose a question directed at the answer. In this task two users were noticed whose intent was not to complete the task properly, but quickly, their input was alternating between a couple of questions disregarding the answer. The quality of the answer passage selections is the same as with the interviews.

In Fig. A4 a histogram of the amount of questions per answer. We can notice that we have way less questions than in the interview case. This lies in the nature of the CrowdFlower task. For the books we asked for 2 workers per task, while for the interview answers it have been 10 workers per task, since the answers from the interviews are assumed to be more frequently asked for. Overall about half of the answers have one or two questions.

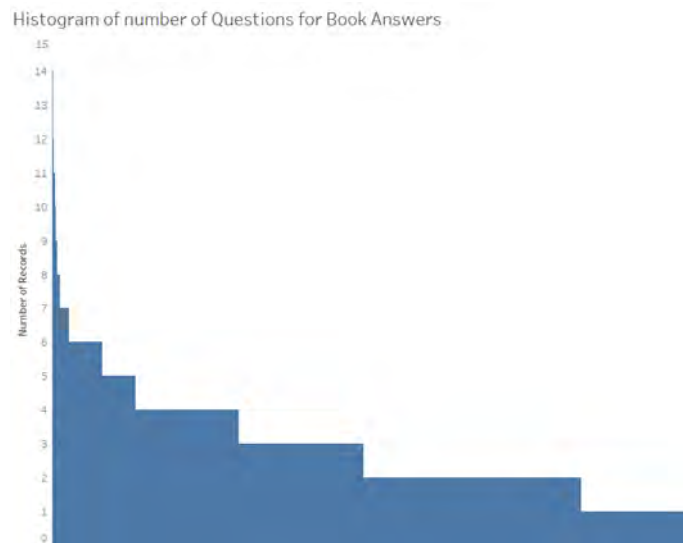


Fig. A4. Histogram of number of questions per answer for the literature data set

Fig. A5 shows a histogram divided by question type (it is filtered by answers who have 3 questions or more, since those are the ones we will use for our experiment first of all to have enough questions and secondly because our interview data set is smaller due to the number of answers fact checked by the museum, so we have to adjust our book data set as well). We can see that for all answers that have five questions or less the question type is always the same, so we have never less than 3 questions per question type.

In Fig. A6 the distribution between the question types is shown. We distinguish here between questions for book answers and questions for interview answers. For the books people asked more about the entity, while having little questions about location. All other types are more or less even. In the interviews people mostly ask about descriptions, while asking other types of questions less. The small percentages of questions about abbreviations were misclassifications, since the classifier of the question type that was used has an accuracy of 80%.

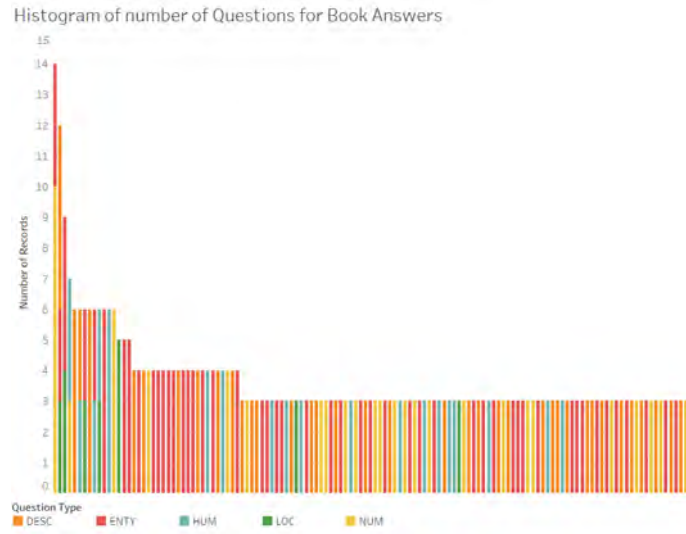


Fig. A5. Histogram of number of questions per answer for the literature data set including question topics

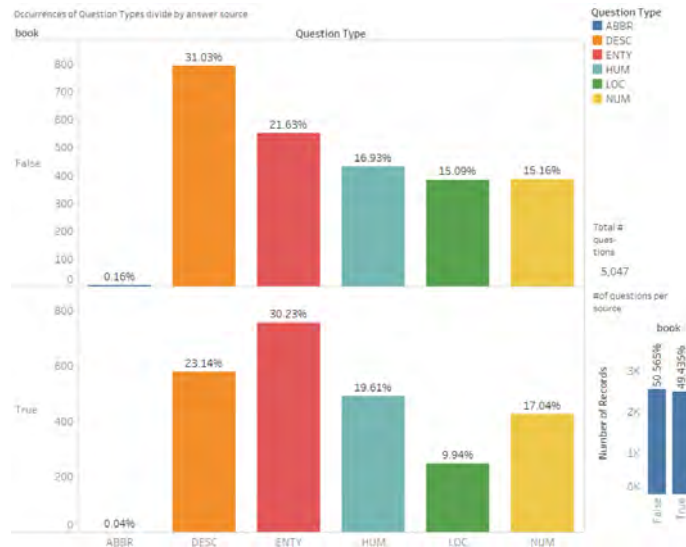


Fig. A6. Occurrence of question type divided by answer source

Appendix B: Diagrams Explaining The SQALPEL System in Detail

In this section the diagrams explaining the SQALPEL system architecture can be found.

In Fig. B1 the path from the raw documents to the database is shown. This also includes the crowdsourcing task that is described in more detail in Appendix A .

Fig. B2 explains the algorithm we have developed to extract paragraphs from a .pdf file of a book.

The retraining process is explained in Fig. B3.

Fig. B4 shows the processing of a user query that is used to create the output that the user sees.

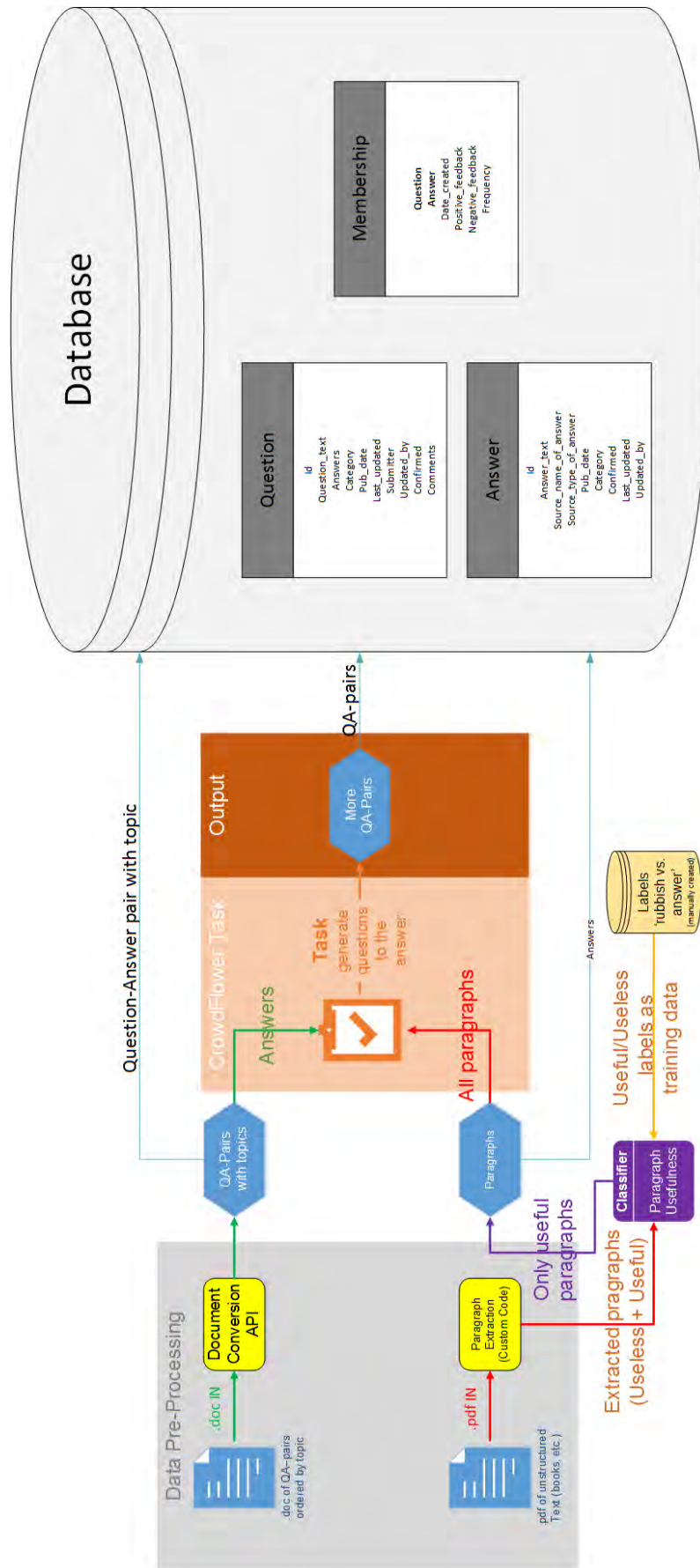


Fig. B1. Diagram explaining how the Documents get to the Database in the SQALPEL system

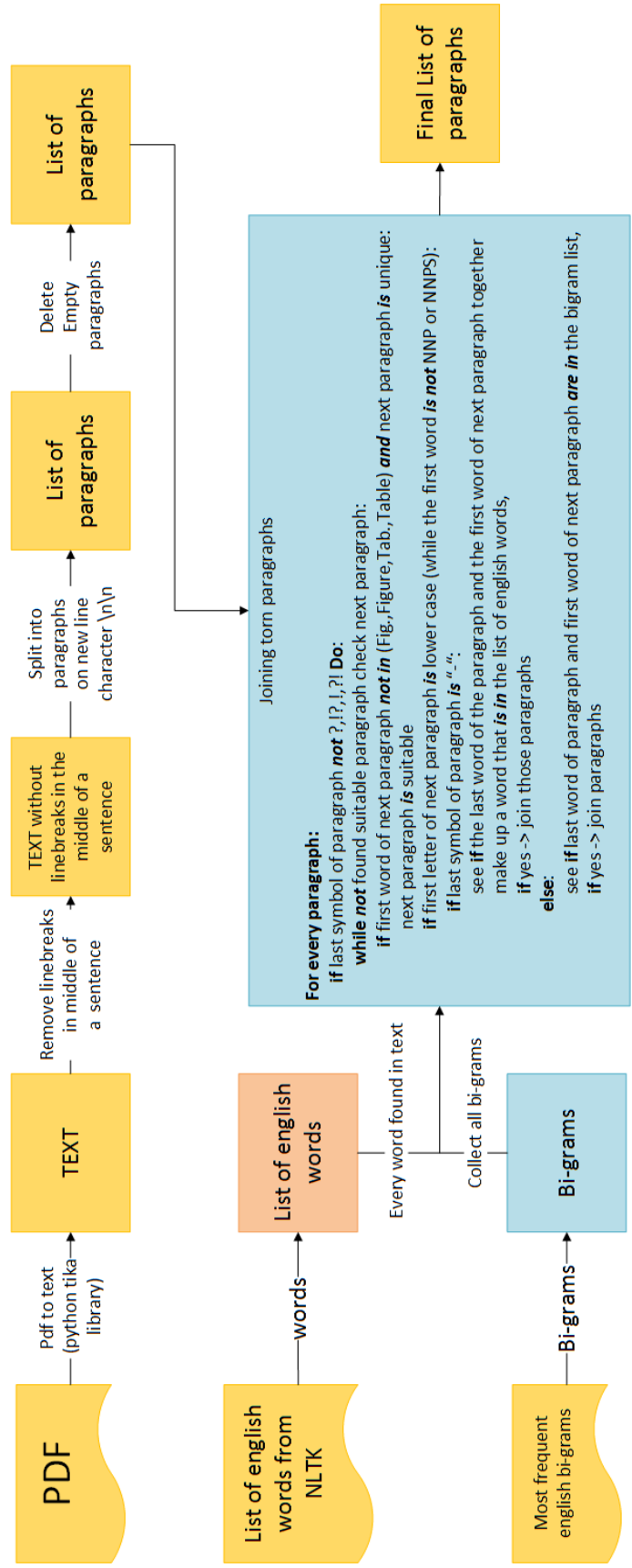


Fig. B2. Algorithm to split a pdf into paragraphs.

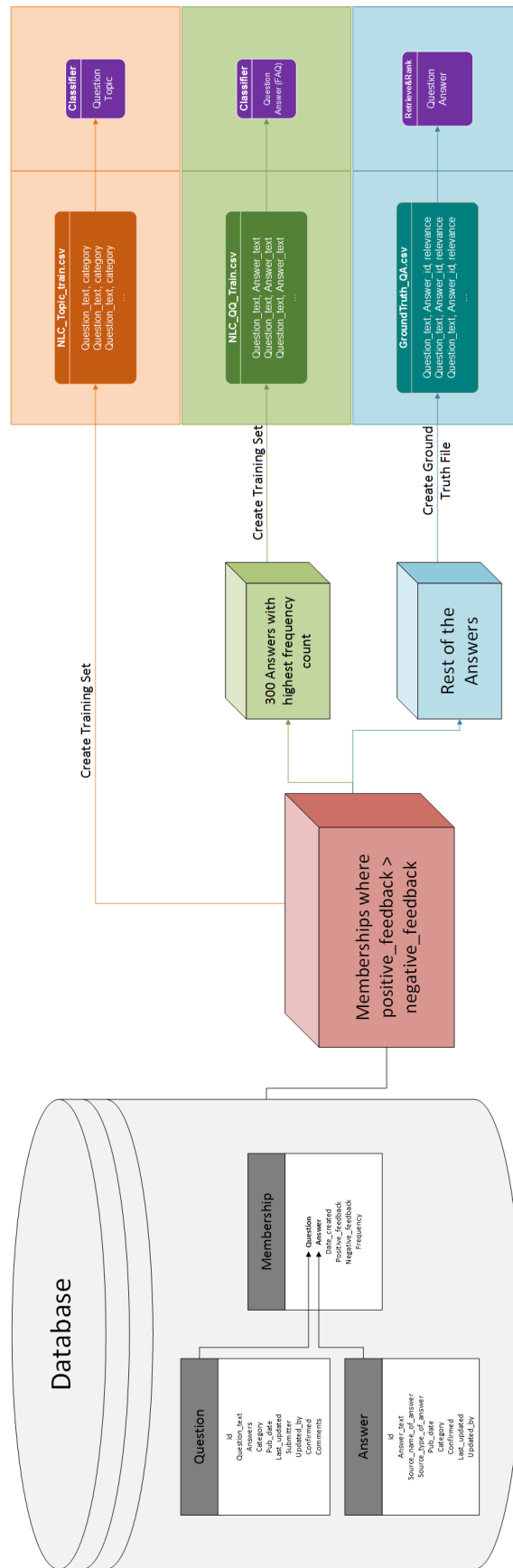


Fig. B3. Retraining the SQUALPEL system.

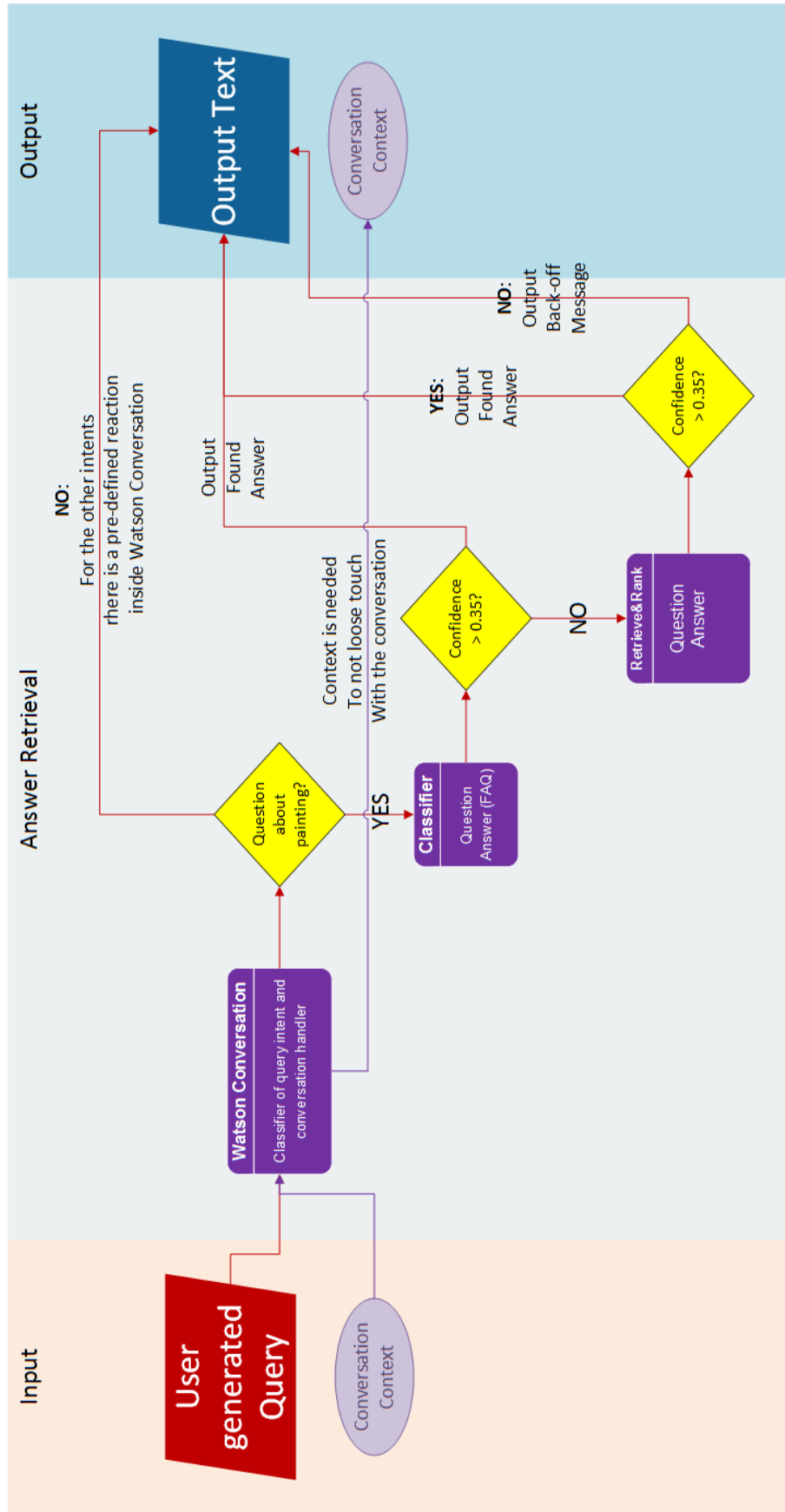


Fig. B4. Diagram explaining the way from a query to the retrieved answer in the SQALPEL system

Appendix C: Screenshots of The Application



Fig. C1. Screenshot of the Starting Screen in the SQALPEL iOS application

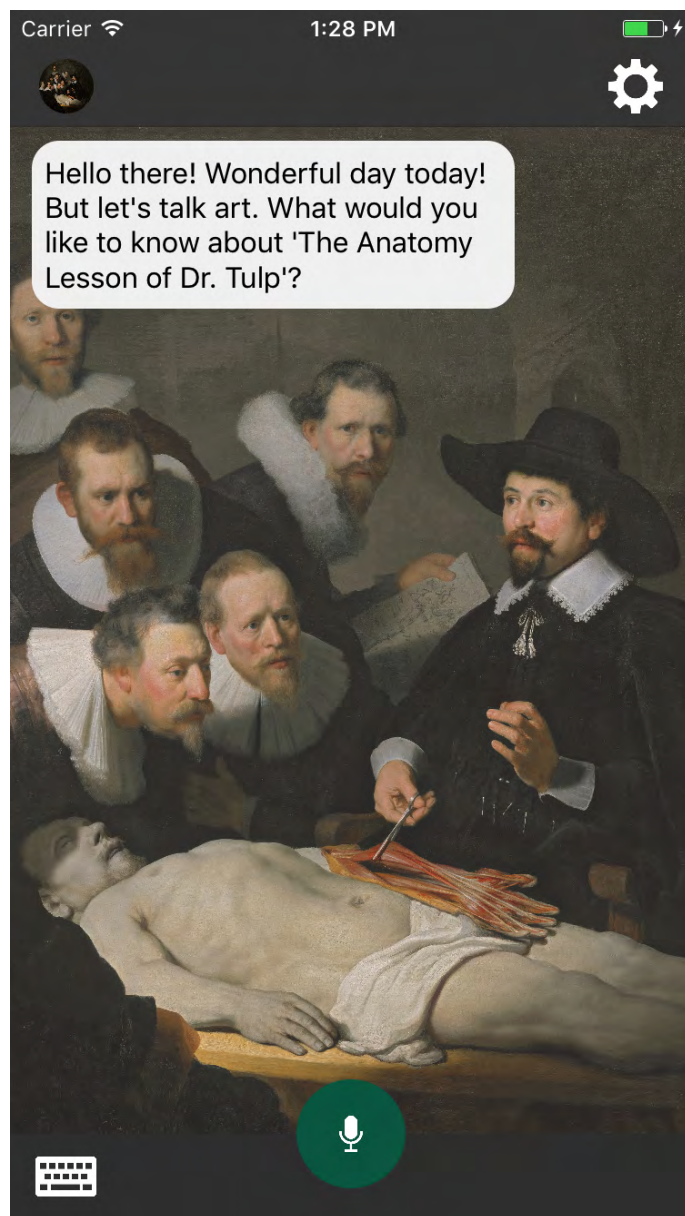


Fig. C2. Screenshot of the Welcome Message in the SQALPEL iOS application



Fig. C3. Screenshot of the chat interface of the SQALPEL iOS application



Fig. C4. Screenshot of the high-resolution painting view in the SQALPEL iOS application

Appendix D: The Project Code And The Experiments Code

The code for the SQALPEL project can be found in the IBM github repository under: https://github.ibm.com/CASBNL/gen_mauritshuis. (An IBM account is needed to access the github page.)

The code for the experiments can be found in the *nikita_exp* branch of the repository: https://github.ibm.com/CASBNL/gen_mauritshuis/tree/nikita_exp.

Appendix E: Classifying Answer Correctness

Measuring MAP requires us to know exactly which answer is correct to which question. This is a problem for us because the data set used in this study is lacking labels, only a fraction of all possible correct answers to each question are annotated. To overcome this problem, we have built a classifier that can classify an answer as correct or not with a high precision.

The classifier we built was inspired by the question answering evaluation metric POURPRE that measures the co-occurrence of words between a given answer and an answer nugget. An answer nugget is defined as a string containing a fact for which the assessor could make a binary decision as to whether a response contained that nugget (Lin and Demner-Fushman, 2006). In the crowdsourcing task aiming to collect questions, users were asked to highlight the part of the answer that answers their question. These annotations we use as the answer nugget.

To test the classifier, we have manually annotated 784 question-answer pairs outputted by the system. Because the data set is too small to appropriately train a classifier we have performed a grid search over 50 boundaries between 0 and 1 and found the boundary of 0.5 to be performing best. With this boundary, the classifier achieves an F1-score of 0.91 on the correct answers and 0.65 on wrong answers. This means that the classifier with this boundary will marginally overestimate the true performance of the system.

In Table E1 examples of question and their answer nuggets are shown to illustrate how the classification works. In the first example all of the words in the answer nugget can be found in the retrieved answer, thus the answer is being classified as correct. The second answer is classified as wrong, because none of the words in the answer nugget are contained in the answer. The third example illustrates the case where some of the words from the nugget are contained in the answer. Here the answer is classified as correct, because 3 out of 4 words are contained in it which is higher than the classification boundary of 0.5.

Table E1. Examples of questions together with their answer nugget and an answer retrieved by the system

Question	Answer Nugget	Answer Retrieved
Who is the man with the hat?	Dr. Tulp	The man with the hat is Dr. Nicolaes Tulp.
Who is responsible for the current location of the painting?	Dutch King William I.	Yes, the Nieuwmarkt in Amsterdam is located next to Chinatown.
What nationality was Rembrandt?	famous Dutch painter Rembrandt.	This masterpiece was made by the renowned Dutch painter Rembrandt in 1632.

The classifier disagrees with our annotations in 112 out of the 784 cases. The main reason for this disagreement is the quality of the annotation: it is either not meaningful and does not contain the answer to the question; it is too long, making the annotation too specific; or it is too short, making the annotation too general. Other reasons include mistakes in our judgment, not considering synonyms, not removing stop words.

Variating the classifier by deleting stop words, applying stemming or considering word importance did not show any improvements. We calculate the MAP of the system after each simulated day on the validation set and the simulation set. We report both numbers to explore whether system performance differs between questions the system has seen during exploitation and questions the system has never seen before.

