Predicting Early User Churn on Duolingo using Machine Learning

Arnoud Verstraaten
STUDENT NUMBER: 1273956

THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN DATA SCIENCE & SOCIETY
DEPARTMENT OF COGNITIVE SCIENCE & ARTIFICIAL INTELLIGENCE
SCHOOL OF HUMANITIES AND DIGITAL SCIENCES
TILBURG UNIVERSITY

Thesis committee:

Dr. A. Alishahi
Dr. G. Nápoles

Tilburg University
School of Humanities and Digital Sciences
Department of Cognitive Science & Artificial Intelligence
Tilburg, The Netherlands
June 2020

Preface

Dear reader,

Thank you for taking the time to read my thesis on predicting user churn on Duolingo. Writing the thesis was a tough process due to the complexity of gathering the data into a workable format and the hard reality of the coronavirus which resulted in the university's library closing. This highly impacted both my progress and motivation.

I can say I am proud and satisfied that in the end I was able to produce this thesis and lived up to the plan I set up for myself back in February.

Best,
Arnoud Verstraaten

Predicting Early User Churn on Duolingo using Machine Learning

Arnoud Verstraaten

This paper explores user churn on the online language platform Duolingo. Given the body of variables provided by Duolingo in the SLAM 2018 Challenge and feature engineered variables, the task is to predict whether the user left the platform within the first two weeks, or not. Machine learning techniques, logistic regression, decision tree, random forest, and support vector machines are applied to a data set containing 2593 learners of English and 2 million exercises over their initial thirty-day period on the platform. The performance of the models is measured in terms of accuracy. Results show that the random forest performs best at 65.5%, followed by decision tree at 64.0%. Logistic regression and support vector machines performed 62.2% and 61.7%, respectively. This work paves ways for future research on user churn prediction on online language platforms.

1. Introduction

Due to a surge in online language learning popularity on applications and on websites, large amounts of user data become available on users learning a new language. This data is of incredible value for online language learning platforms and researchers. In the 21st century, data is viewed by experts as 'the new oil': valuable, but if unrefined it cannot really be used. The data derived from the users create opportunities to improve platforms by creating more personalized instructions for the users. These personalized additions cause improvements in the user experience, and keeps them engaged in learning the language on their platform.

Duolingo is such a language platform, where users can learn languages for free. Users can purchase a membership each month for a premium service that removes advertisements and delivers offline access to their educational material. The platform is the world's largest online language learning provider with over 300 million users from countries all over the world. The core part of Duolingo and its AI strategy is to provide users with a human-to-human learning experience. Maintaining an engaging and immersive experience for the user is crucial in motivation the learner in returning to the platform, which is especially challenging for the platform in teaching the language on a touch screen, or on a desktop.

Aiming at keeping the online language learners engaged on their platform, Duolingo's AI department launched bots and delivered short messages within the application to boost the user's engagement. Similarly, bots could potentially be developed that use machine learning models that accurately predict users leaving the platform. Duolingo could create and execute a strategy that keeps the user motivated in studying a language on its platform, thereby minimizing the number of users from churn. As a result, the language platform could retain users on their platform and users will remain motivated to use the platform and progress in learning the language they want to learn.

Customer churn is widely discussed in academic papers and in businesses, e.g. in the works of Almana, Aksoy, and Alzahrani (2014) and Farquad, Ravi, and Raju (2014). Often the scope of the work in customer churn is in field such as banking (Min and Jeong 2009) or telecommunications (Almana, Aksoy, and Alzahrani 2014). However, this thesis will research a different field and investigate early churn of users learning a language learning on an online platform.

The models developed in this thesis use machine learning techniques to predict early user churn on Duolingo, specifically users churning within the first two weeks of signing up on the platform. The data set used in this thesis is from Duolingo's platform and was released by their AI team in 2018 during the Second Language Acquisition Modeling 2018 Challenge. Logistic regression, decision tree, random forest, and support vector machine will be constructed and compared to each other with the accuracy evaluation measure, where the model that scores highest on the test set is considered to be the best performing model. The models use the features provided by Duolingo and also use several feature engineered variables converted from exercise level to user level.

The main contributions of this work are the development of a churn prediction model that assists language platforms in predicting users who are probable to churn. Given the Duolingo's SLAM 2018 data set, the main research question of the thesis is: What machine learning model performs best in predicting user churn on online language learning platforms?

To answer this question, exploratory data analysis will be performed on the data set to understand users' learning characteristics. The last day of activity for the 2593 users is of particular interest to understand when the users churn. Moreover, finding patterns when users stop being active to determine churn within the first two weeks is crucial. Therefore, this thesis also aims to construct new features from the exercise level data and convert them into user level data in order to process them into the machine learning models.

Overall, all four models clearly outperform the baseline model of 52.3%. The random forest model performs best out of the four machine learning models with an accuracy of 65.5%, followed by the decision tree at 64.0%, logistic regression at 62.2%, and support vector machines at 61.7%, respectively.

2. Related Work

2.1 Duolingo SLAM 2018

In Duolingo's SLAM 2018 Challenge[1] fifteen teams competed to create a model that correctly predicts an error a user makes at an arbitrary point in time given the users' past mistakes and user characteristics. The data contains a body of seven million words produced by over 6,000 users from three different language tracks: Spanish, English, and French over a period of thirty days. The teams were encouraged to engineer new features from the data set and to construct and optimize different machine learning models.

The paper by Settles et al. (2018) combined the works of the fifteen teams and summed up the findings of the papers written by eleven of the teams in their report. The outcome of the paper indicates that feature engineering had a much smaller impact on

[1] https://sharedtask.duolingo.com/2018.html, accessed on 10-05-2020.

model performance than the selected learning algorithm in modeling a user's mistake, except for the feature average time spent per exercise. Moreover, word frequency, which is the amount of time a word was revealed to the user, was popular but was unable to yield significant improvement (Settles et al. 2018).

2.2 Customer churn

Customer churn, the rate at which customers seize doing business with a company, is one of the most important concerns for large companies since it directly affects the revenue streams of firms. Feature engineering is one of the most difficult and complex processes in building predictive models for customer churn (Ahmad, Jafar, and Aljoumaa 2019).

According to Gallo (2014), acquiring a new customer is five to twenty-five times more costly than retaining an existing one. Therefore, understanding the relevant factors that identify customer churn is important in developing a strategy to reduce churn. Moreover, improving customer retention appears essential for the longevity of a firm's business (Gallo 2014). Research on the subject of churn prediction with machine learning models is widely conducted in both academia and business, e.g. in papers like Xie et al. (2009) and Neslin et al. (2006).

2.3 Logistic regression

Logistic regression is a classic statistical model that predicts the probability of an outcome and is applicable in binary classification settings, e.g. predicting churn (Chandrayan 2019). With the threshold boundary, a clear classification can be given to the input. For example, in case the probability that a user is about to churn from the platform is greater than 0.5, then the user will be classified as a churner (Chandrayan 2019). In other cases, the user remain on the platform (Chandrayan 2019).

Logistic regression is often applied in customer churning predictions (Vafeiadis et al. 2015). In their paper, Vafeiadis et al. (2015) compared different models in customer churn predictions. Monte Carlo simulations were performed on different models without boosting them under different settings. The results of applying logistic regression are poor when compared against other models like neural networks, decision trees, and support vector machines (Vafeiadis et al. 2015). Logistic regression scored comparatively low with an accuracy of 86% (Vafeiadis et al. 2015).

The advantage of using logistic regression is that the model by itself is simple and efficient (Chandrayan 2019). The variance is low, which makes generalizations from the training data to new data more accurate (Singh 2018). That said, logistic regression does not perform well with categorical features (Singh 2018).

2.4 Support vector machines

The support vector machine method was introduced by Boser, Guyon, and Vapnik (1992). Support vector machines optimize the distance in the training data features and create the decision boundary for the different labels (Boser, Guyon, and Vapnik 1992). These boundaries are created in multi-dimensional spaces, solving quadratic optimization problems into groups (Dreiseitl and Ohno-Machado 2002).

According to Vafeiadis et al. (2015) and Gaspar, Carbonell, and Oliveira (2012), the selection of the best kernels or combinations of kernels is crucial to the technique's performance. Both papers argue that kernel selection can be substantially improved and that in the future the method will be improved. Using different kernel functions, varying degrees of nonlinearity and flexibility can be included in the model (Dreiseitl and Ohno-Machado 2002). The classification accuracy in some cases improves significantly due to using different kernels, however, in other cases it has no significant effect (Vafeiadis et al. 2015).

Support vector machines hold high potential against traditional approaches due to their scalability, faster training and running times (Zhao et al. 2005) and support vector machines consistently outperforms decision trees and in some cases artificial neural networks for customer churn prediction (Vafeiadis et al. 2015).

Support vector machines' main limitation is the fact that it is not transparent how the model comes to its classification and operates in a 'black box'. The trained algorithm from the training set is incomprehensible to human beings (Farquad, Ravi, and Raju 2014). Mechanisms are being developed to retrieve information from the algorithm to convert them into rules to reveal the decision process and make it visible to customer relationship management. Overall, the extracted rules are neither exclusive nor exhaustive which results in rules that are incomplete and inconsistent for the classification of new instances (Farquad, Ravi, and Raju 2014). Additionally, a disadvantage of support vector machines is that each group label is decided by the decision boundary, and no probability is provided for the support vector machines' labeling (Dreiseitl and Ohno-Machado 2002).

The support vector machines algorithm is difficult to utilize with large data sets since the training time with the algorithm is computationally exhaustive. However, solutions are available to overcome the issues (Tsang, Kwok, and Cheung 2005). Moreover, the model works well in binary classification problems (Hsu and Lin 2002), e.g. churn prediction.

In their paper, Brânduşoiu, Toderean, and Beleiu (2016) compared several methods in terms of accuracy within the telecommunications industry, including neural networks, and Bayesian networks and support vectors machines. Their results indicated that support vector machines outperformed both Bayesian networks and neural networks in terms of accuracy, each model achieving over a 99% accuracy, albeit by a small margin. The results suggest that these four methods all perform well, however, the support vector machines outperforms the others slightly (Brânduşoiu, Toderean, and Beleiu 2016).

2.5 Decision tree and Random forest

Decision trees, random forests in churn management are discussed in many works, e.g. in the works of Hassouna et al. (2016) and Hung, Yen, and Wang (2006). The decision tree is a predictive model that is suitable in binary classification settings (Min and Jeong 2009). The algorithm scored 76.8% accuracy, outperforming logistic regression at 70.7% accuracy in the binary prediction task of determining if a company goes bankrupt, or not (Min and Jeong 2009).

Another study by Xie et al. (2009) found that the application of random forests in the banking world yielded superior results over artificial neural networks, decision trees and support vector machines.

In the paper by Hung, Yen, and Wang (2006), the authors evaluated in their work the prediction accuracy of churn in the telecommunications industry with one single decision tree and a random forest. The random forest method showed significantly better results in the telecommunications industry over the decision tree (Hung, Yen, and Wang 2006).

Decision tree algorithms perform increasingly poorly as the data set grows larger (Almana, Aksoy, and Alzahrani 2014). To mitigate this, researchers use random forest over decision trees to create robust and more accurate results (Almana, Aksoy, and Alzahrani 2014). The random forest outperforming decision trees is particularly pronounced in the prediction of imbalanced classes (Wei and Chiu 2002). Random forest outperforming decision trees is consistent according to the works of (Hung, Yen, and Wang 2006) and (Wei and Chiu 2002).

3. Experimental Setup

3.1 Dataset Description

The dataset is from Duolingo's challenge '2018 Duolingo Shared Task on Second Language Acquisition Modeling' (SLAM) and contains users' activity while learning a new language doing exercises for the first thirty days on the platform. The data is openly available on Harvard's Dataverse, downloadable in a tar.gz file, which is a type of file with a combination of TAR packaging followed by a GNU zip (gzip) compression. Three language tracks are available: English learners for Spanish speakers, Spanish learners for English speakers, and French learners for Spanish speakers. Out of the three tracks, the English learners for Spanish speakers track is used in the analysis due to computational limitations since over 2 million observations are present in the data set.

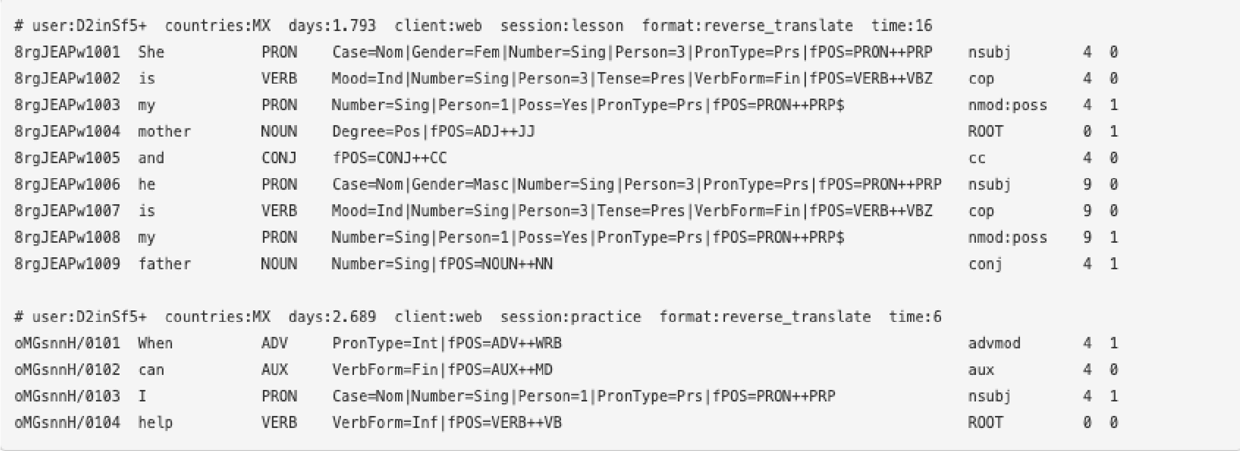
The data format that is used to retain the data is inspired by the Universal Dependencies CoNNL-U format [2]. The annotations are encoded in plain text files (UTF-8, normalized to NFC, using only the LF character as a line break, including an LF character at the end of file) with three types of lines: (1) word lines containing the annotation of a word/token in 10 fields separated by single tab characters, (2) blank lines marking sentence boundaries, and (3) comment lines starting with a hash (#). The student exercises are represented by multiple lines, separated by a blank line between the exercises. One token per line with exercise-level metadata on the first line. An example is illustrated in the image below[3].

On the first line of each exercise group the following data about the user, the session, and the exercise is kept[4]. User: a B64 encoded, 8-digit, anonymized, unique identifier for each student (may include / or + characters). Countries: a pipe (|) delimited list of 2-character country codes from which this user has done exercises. Days: the number of days since the student started learning the language on Duolingo. Client: the student's device platform: android, ios, or web. Session: the session type (one of: lesson, practice, or test; explanation below). Format: the exercise format (one of: reverse translate, reverse tap, or listen; see figures below). Time: the amount of time (in seconds) it took for the student to construct and submit their whole answer (note: for some exercises, this can be null due to data logging issues)

The other lines in the group represent each word in the correct answer that is most similar to the student's answer, one token per line, arranged into seven columns separated by whitespaces[5] which includes: a unique 12-digit ID for each token instance, the word, part of speech in Universal Dependencies (UD) format, morphological features in

[2] https://universaldependencies.org/docs/format.html, accessed on 13-05-2020.
[3] https://sharedtask.duolingo.com/2018.htmltask-definition-data, accessed on 13-05-2020.
[4] https://sharedtask.duolingo.com/2018.htmltask-definition-data, accessed on 13-05-2020.
[5] https://sharedtask.duolingo.com/2018.htmltask-definition-data, accessed on 13-05-2020.

Figure 1
Duolingo’s data set format in CoNLL-U, including user-specific metadata on the first line, and word level data on the subsequent lines.

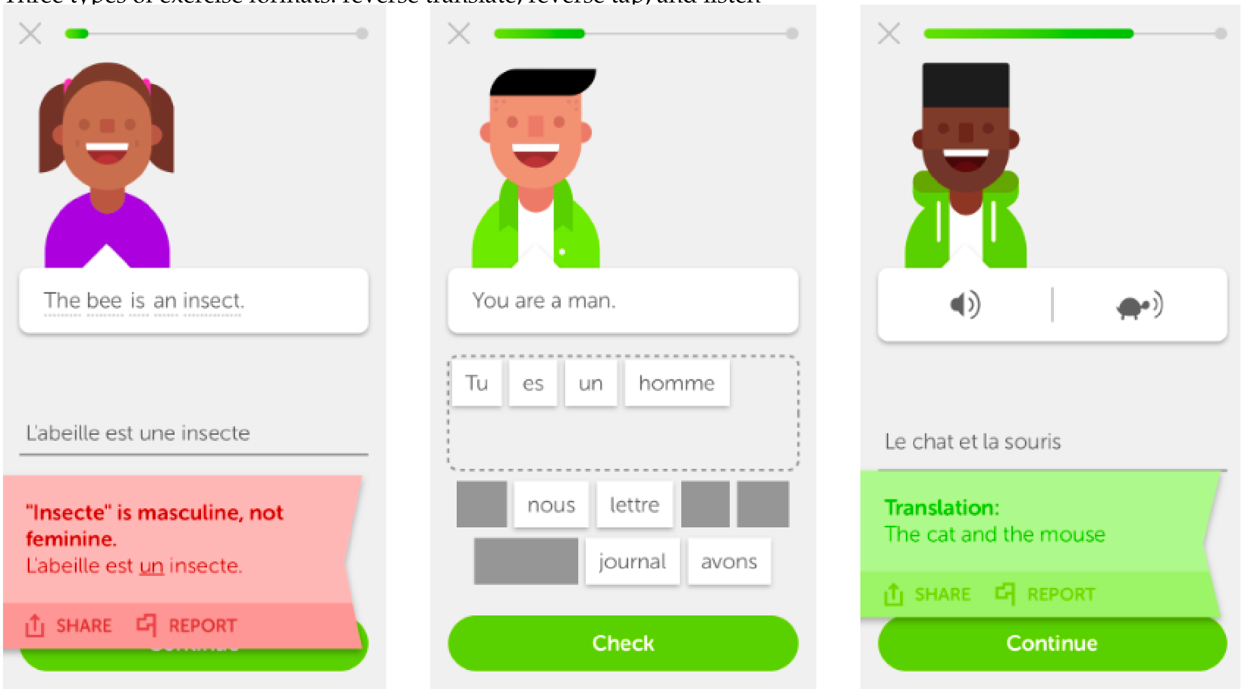


```
# user:D2inSf5+  countries:MX  days:1.793  client:web  session:lesson  format:reverse_translate  time:16
8rgJEAPw1001  She         PRON   Case=Nom|Gender=Fem|Number=Sing|Person=3|PronType=Prs|fPOS=PRON++PRP   nsubj       4  0
8rgJEAPw1002  is          VERB   Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin|fPOS=VERB++VBZ   cop         4  0
8rgJEAPw1003  my          PRON   Number=Sing|Person=1|Poss=Yes|PronType=Prs|fPOS=PRON++PRP$             nmod:poss   4  1
8rgJEAPw1004  mother      NOUN   Degree=Pos|fPOS=ADJ++JJ                                                ROOT        0  1
8rgJEAPw1005  and         CONJ   fPOS=CONJ++CC                                                          cc          4  0
8rgJEAPw1006  he          PRON   Case=Nom|Gender=Masc|Number=Sing|Person=3|PronType=Prs|fPOS=PRON++PRP  nsubj       9  0
8rgJEAPw1007  is          VERB   Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin|fPOS=VERB++VBZ   cop         9  0
8rgJEAPw1008  my          PRON   Number=Sing|Person=1|Poss=Yes|PronType=Prs|fPOS=PRON++PRP$             nmod:poss   9  1
8rgJEAPw1009  father      NOUN   Number=Sing|fPOS=NOUN++NN                                              conj        4  1

# user:D2inSf5+  countries:MX  days:2.689  client:web  session:practice  format:reverse_translate  time:6
oMGsnnH/0101  When        ADV    PronType=Int|fPOS=ADV++WRB                                             advmod      4  1
oMGsnnH/0102  can         AUX    VerbForm=Fin|fPOS=AUX++MD                                              aux         4  0
oMGsnnH/0103  I           PRON   Case=Nom|Number=Sing|Person=1|PronType=Prs|fPOS=PRON++PRP              nsubj       4  1
oMGsnnH/0104  help        VERB   VerbForm=Inf|fPOS=VERB++VB                                             ROOT        0  0
```

UD format, dependency edge label in UD format, Dependency edge head in UD format, and lastly, the label to be predicted (0 or 1).

Figure 2
Three types of exercise formats: reverse translate, reverse tap, and listen[6]



The bee is an insect.

L'abeille est une insecte

"Insecte" is masculine, not feminine.
L'abeille est un insecte.

You are a man.

Tu es un homme

Le chat et la souris

Translation:
The cat and the mouse

(a) reverse_translate (b) reverse_tap (c) listen

Figure 2, illustrates the three types of exercise formats that are present on the Duolingo platform for the users. The first exercise format is reverse translate, in which the user reads a prompt written in the language they master, e.g. native language, and translates the prompt into the language the user is studying. The second format is reverse tap, which is a simpler version of the previous format since the user needs to construct the answer from provided words that include distraction words. The last format is listen, where the user listens to a prompt in the language they are learning and has to translate the sentence back into the language they master. The exercises can have more correct answers, in some cases a couple of thousand. This is because of a number of reasons, including a multitude of synonyms, homophones, formality, and more. Duolingo uses a method called FSTs, which aligns the user's input to the most similar correct answers in the set of acceptable answers provided by Duolingo. In the figure above, corrective feedback is provided when the user's answer is incorrect.

3.2 Preprocessing

The data provided by Duolongo's AI team was organized in the aforementioned CoNNL-U format. The code is written to extract load and read the data. The meta information of an exercise is concatenated with the exercise level data. The 2 million observations are then combined in a NumPy array by writing a code that splits the values from the CoNLL-U formatted data set. Originally, the data set had no missing values, however, logging issues occurred in the time feature (Settles et al. 2018) causing these values to be converted into negative values. The negative values are replaced by NaN's in the Pandas data frame and are treated as missing values. Settles et al. (2018) reports that no additional anomalies were identified in the data set and thus the assumption is made that no logging issues have occurred. The categorical features are converted to dummy variables in order to make the features compatible with the machine learning models. The Pandas package (McKinney 2012) is used to create the dummy variables for variables country, platform, session, and format.

3.3 Feature engineering

Three new variables are constructed from the existing data set. These variables are constructed from the exercise-level data and converted into user-level data in order to process the data into the machine learning algorithms.

First, mean accuracy per user is constructed by looping through all the exercises done for each of the 2593 users in the data set. It is calculated by summing the correct answers by the user, and dividing the outcome by the total number of exercises the student participated in. The feature captures the effect of discrepancy in learning speed and in accuracy (Tomoschuk and Lovelett 2018).

Second, the number of exercises feature is the number of exercises each user has completed in total and is calculated by looping through each exercise, determining to which user the exercise belongs to and attaching the number of loops done per user.

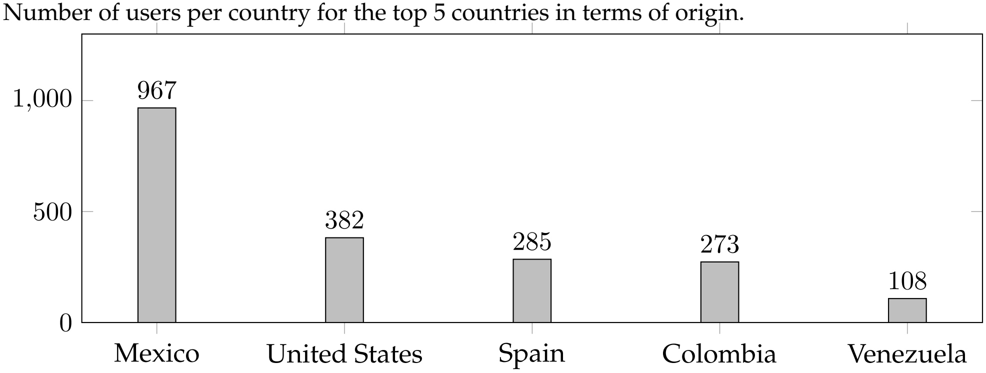
Lastly, average time spent per exercise. Average time spent per exercise measures the average time the user takes to provide the answer to the exercise. It is calculated by summing the time the user spent answering exercises and dividing the total sum by the number of exercises the user has completed.

3.4 Exploratory data analysis

The day the user was last active during the thirty day period is relevant to investigate. On the first day of the period, most users continue to use the platform. From the second day through day twenty, the users' last activity is quite evenly distributed. After day twenty, we see a decline in user activity, indicating that the users in the sample did not use the platform during those days. Interestingly, almost no activity is present beyond the twenty-fifth day.

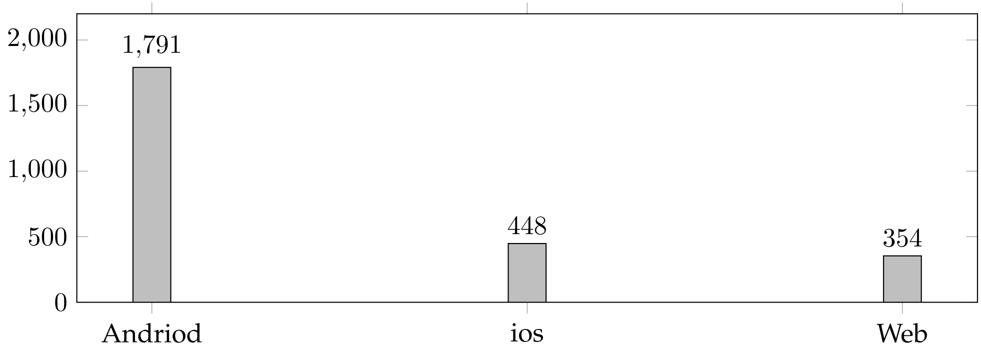
In figure 3, the five most common countries for English learners are from Mexico with 967 users, the United States with 382 users, Spain with 285 users, Colombia with 273 users, and finally, Venezuela with 108 users. The majority of the users are from a country where the native language is Spanish. I expect that users from the US are mostly Mexican immigrants, or other Spanish speaking immigrants, trying to learn the English language.

Figure 3
Number of users per country for the top 5 countries in terms of origin.



In figure 4, the number of users who used android devices to learn English on the platform is 1791, the number of users who used ios devices is 448, and the number of users who used their desktop or laptop is 354. From the figure, it becomes evident that mostly android users are present in the data set. This large discrepancy might be due to the affordability of android over ios in the countries that the users are from, or simple preference. Moreover, it appears that users generally use the smartphone application over the web application.

Figure 4
Number of users who use an android device, ios device, or who uses a device that accesses the web.



In figure 5, the number of users who mostly used reverse translate exercises is 1161. The number of users who mostly used listen exercises is 724. The number of users who mostly used reverse tap exercises is 708. The amount of users mostly doing reverse translate exercises is larger than those mostly doing listen or reverse tap exercises.

Figure 5
Number of users with most exercises in types of exercises: reverse translate, listen, and reverse tap.

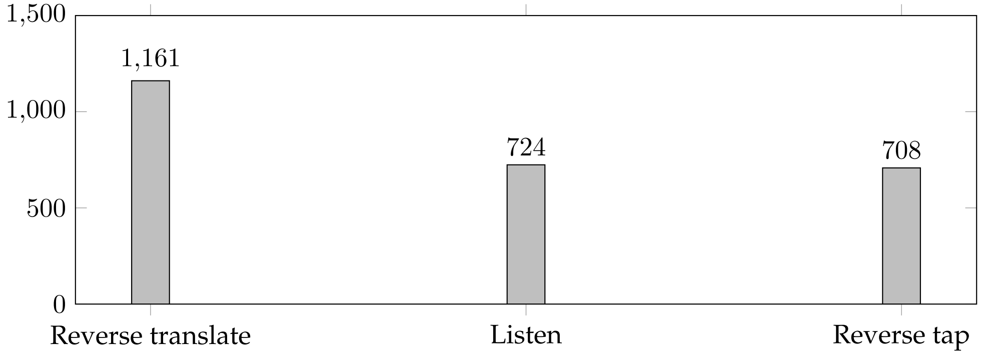


Table 1
Descriptive statistics for the quantitative features: last day of activity on the platform, time spent on the platform (in seconds), number of exercises completed, mean accuracy, and mean time spent per exercise (in seconds).

Feature	Count	Mean	Std	Min	Max	Median
Days	2593.00	11.41	6.54	0.00	28.00	11.00
Time	2593.00	3649.21	7544.22	0.00	103828.60	15606.00
Exercises	2593.00	1011.55	651.41	103.00	8894.00	835.00
Accuracy	2593.00	87.72	6.75	26.46	99.19	89.29
Mean time	2593.00	36.16	78.32	0.00	1388.11	17.34

In Table 1, the table with the descriptive statistics for the quantitative features are presented. Each feature has no missing values with a total amount of observations of 2593.

The feature days has a mean of 11.41, which means that, on average, the user's last activity was registered 11 days after using the platform. The minimum value is 0.00, indicating that some users left the platform immediately after signing up. The maximum value is 28, indicating that the maximum number of days users were active is 28. The standard deviation is 6.54.

The feature total time spent on the platform has a mean of 3649.21 seconds. This means that the user, on average, was active for an hour during their time on the platform, the standard deviation is quite high at 7544.22, indicating large differences between users. The maximum amount of time a user spent on the platform is 103828.60 seconds, which is close to 30 hours. The median time spent is a little over 4 hours.

The feature number of exercises has a mean of 1011.55, indicating that the user, on average, completed 1012 exercises. The standard deviation is 651.41. The minimum amount of exercises completed is 103, the maximum is 8894, and the median is 835.

The feature accuracy has a mean of 87.72, indicating that, on average, the user provides the correct answers to the exercise 87.72 times out of 100. The standard deviation is relatively low at 6.75. The minimum accuracy of a user is 26.46%, the maximum accuracy is 99.19%, and the median accuracy is 89.29%.

Finally, the feature mean time spent per exercise is 36.16 seconds, indicating that the user, on average, spent 36.16 seconds per exercise. The standard deviation is quite high at 78.32, suggesting that users spent varying amounts of time to provide an answer to the exercises. The minimum mean time spent per exercise is 0, the maximum is 1388.11, and the median is 17.34.

3.5 Models

The Zero Rule algorithm will be used as the baseline. The algorithm predicts the label that is the most common in the distribution for each user. Therefore, the algorithm predicts that the user churns for each user, which results in a baseline of 52.3% due to the label distribution, where 52.3% in the sample of users on the platform is labeled early churner.

The first model is a simple logistic regression. It models the probability of output in terms of input and has no model specific hyperparameters.

The second model is a decision tree. The function to measure the split quality that is used in the model is Gini impurity. The default settings from the Sklearn (Pedregosa et al. 2011) are used. The tree is pruned to a maximum depth of 5. The random state is set to 42.

The third model is the random forest. The function to measure the split quality that is used in the model is the Gini impurity. The number of trees used in the forest is set to 100, which is in line with the findings of Bernard, Heutte, and Adam (2009). The tree is pruned with a maximum depth of 5. The random state is set to 42.

The final model is the support vector machine. The hyperparameters have been tuned for the model. The parameter c is set to 1000 since this provided the highest accuracy on the test set. The kernel has been set to RBF. Gamma has been set to 'scale' in the scikit-learn environment. The random state is set to 42.

3.6 Methods

For the data set a train and test set split needs to be made in order to feed the models and let them train. The 2593 observations will be randomly split by using the sklearn package, where 80% is placed in the train set and 20% is placed in the test set.

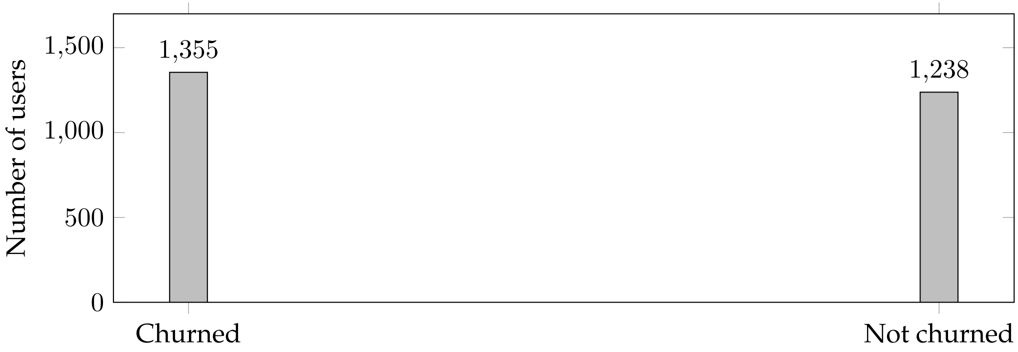
The quantitative data needs to be scaled. The units in the quantitative features of the data are different. The unit of time is in seconds, whereas accuracy, the features of exercise formats, and average time spent per exercise are fractions and seconds, respectively. To overcome the issues in training the machine learning models it is better to scale the data. Consequently, the features number of exercises, total time, accuracy, and mean exercise time will be scaled.

The prediction problem of my model will be classification. In this thesis, accuracy will be the evaluation method to distinguish performance between the four different models because this is common practice in the papers discussed in the related works section. Accuracy measures the rate of the correctly classified instances of the two classes. The evaluation metric is a method to quantify the performance of a machine learning model. The metric is a widely used evaluation metric in churn prediction across sectors in a binary setting, e.g. in (Chandrayan 2019) and Vafeiadis et al. (2015). The model with the highest accuracy rate on the test set is considered to have performed best in predicting user churn on the language platform.

The ground labels are obtained by labeling all the users for whom the last activity on the platform was within the first fourteen days out of the thirty-day period, i.e. the user is considered to have churned early the platform if the user was not active after day fourteen and considered to not have churned early if the user was still active after the fourteen days. Therefore, the task of the prediction models is to predict if the user was last active in the first fourteen days, or the user was last active on the platform on day fifteen or later. For example, the user's last documented activity was on day twelve; the model would be correct if it predicts that the user would have churned the platform early. In case the user's last documented activity was on day twenty; the model would be correct if it predicts that the user would not have churned the platform early.

In figure 6, the number of users that churned early during their initial fourteen days on the platform is 1355, and the number of users that did not churn during this period is 1238. Overall, this leads to a balanced distribution in the target labels since both classes are close to equally represented in the data set.

Figure 6
The number of users who churned within the first two weeks, and the number of users who did not.



3.7 Software

Programming language Python is used (Sanner et al. 1999), specifically version 3.0 in the Google Colaboratory environment (Bisong 2019). This environment is chosen for preprocessing computational purposes since large amounts of data need to be processed and the service provided by Google allows access to fast GPUs.

The pandas library is used as a preprocessing, data analysis, and manipulation tool. The Matplotlib library is used for creating plots, to be exported to LaTex format. The Sklearn package is used to create machine learning models. Finally, the different python libraries are the following: Pandas (McKinney 2012), Google Colaboratory (Bisong 2019), Matplotlib.pyplot (Tosi 2009), NumPy (Walt, Colbert, and Varoquaux 2011), and lastly, Sklearn (Pedregosa et al. 2011).

4. Results

The result of the comparison of different machine learning algorithms is shown in Table 2. The left side of the table shows what model was used. The right side of the table shows the accuracy score of the model on the training set and the test set.

Table 2
Best performing models classifying churn on Duolingo. Accuracy on training and test set, respectively.

Models	*Accuracy* score	
	Training set	Test set
Logistic Regression	0.661	0.622
Decision Tree	0.662	0.640
Random Forest	0.678	**0.655**
Support Vector Machine	0.740	0.617

The logistic regression model obtained an accuracy of 66.1% on the training set and an accuracy of 62.2% on the test set. With the 62.2% accuracy on the test set, the model outperformed the baseline by 9.9%. The decision tree model obtained an accuracy of 66.2% on the training set and an accuracy of 64.0% on the test set. With the 64.0% accuracy on the test set, the model outperformed the baseline by 11.7%. The random forest model obtained an accuracy of 67.8% on the training set and an accuracy of 65.5% on the test set. With the 65.5% accuracy on the test set, the model outperformed the baseline by 13.2%. The support vector machines model obtained an accuracy of 74.4% on the training set and an accuracy of 61.7% on the test set. With the 61.7% accuracy on the test set, the model outperformed the baseline by 9.4%.

Overall, all models clearly outperform the baseline models. The random forest model performs best out of the four machine learning models with an accuracy of 65.5%, followed by the decision tree at 64.0%, logistic regression at 62.2%, and support vector machines at 61.7%, respectively.

5. Discussion

The main goal of this thesis was to investigate what machine learning models are applicable to predict early user churn on online language platform Duolingo, and subsequently, investigate which one performs best. Some new features were constructed from the raw data set since the data set had to be transformed into user-specific data in order to feed the algorithms.

The logistic regression model scored relatively better than expected with an accuracy of 62.2% on the test set, outperforming the baseline by 9.9%. The model performs worse than the decision tree and random forest models by 1.8% and 3.3%, respectively. These findings are in line with the expectations of Vafeiadis et al. (2015), where the author stated that the logistic regression model generally performs poorly compared to decision trees and random forest models. The model performs 3.9% lower on the test set compared to the training set, suggesting that the model generalizes well. This finding is in line with the findings of Singh (2018), in which he stated that the low variance of the model makes generalizations more accurate.

The decision tree model scored as expected with an accuracy of 64.0% of the test set, outperforming the baseline by 11.7%. The model outperforms logistic regression and support vector machines. However, the decision tree model underperforms the random forest model by 1.5%. This finding is in line with the findings of Xie et al. (2009) in the banking industry in which the decision tree underperformed the random forest model.

Moreover, the model is quite robust with a discrepancy of 2.2% between the training set and the test set. This is in line with the findings of Almana, Aksoy, and Alzahrani (2014), in which he argues that decision trees perform well as long as the data set is relatively small.

The random forest model outperformed the other machine learning algorithms by a substantial margin with an accuracy of 65.5% on the test set. The random forest method outperformed the decision tree by 1.5%. This finding was also found in the work of Hung, Yen, and Wang (2006) within the telecommunications industry, in which random forests consistently outperform decision trees.

Surprisingly, the support vector machine performed the worst out of the four models with an accuracy on the test set of 61.7%, outperforming the baseline by 9.4%. This is surprising since the model is supposed to work well in a binary classification setting (Hsu and Lin 2002) and because support vector machines consistently outperform decision trees and for customer churn prediction (Vafeiadis et al. 2015).

Moreover, the accuracy on the training data is high. However, the model overfitted and failed to generalize to the test data. This was suggested in the work of Farquad, Ravi, and Raju (2014), where he argued that extracted rules are neither exclusive nor exhaustive which results in rules that are incomplete and inconsistent for the classification of new instances. Moreover, the selection of the best kernels or combinations of kernels is crucial to the technique's performance (Vafeiadis et al. 2015). Consequently, future research might benefit from improved kernel selection.

Ultimately, the data set used in this thesis to predict user churn on language platforms is limited. The data that is used consists of exercise level features and were converted into user-level features. Future research can benefit from more complete user-specific data, similar to data sets provided to researchers by the telecommunications sector (Brânduşoiu, Toderean, and Beleiu 2016) and banking (Min and Jeong 2009), which consist of user-specific features such as gender, age, married or not married and more. In those two fields, accuracy rates of over 90% have been achieved by the same models used as this thesis, except for logistic regression. That said, the models in this thesis have shown promising results by sharply outperforming the baseline using a limited churn-specific data set.

6. Conclusion

In this thesis, early user churn prediction on online language learning platforms is introduced. The main aim was to construct machine learning algorithms to predict early user churn on online learning language platform Duolingo.

The data set provided by Duolingo in the SLAM 2018 challenge is used, taking the original features and feature engineered variables. The data set contained 2593 learners of English and 2 million exercises over their initial thirty-day period on the platform. The task was to predict whether the user left the platform within the first two weeks, or not.

Machine learning techniques, logistic regression, decision tree, random forest, and support vector machines are applied to predict early user churn. The performance of the models is measured in terms of accuracy. Results show that the random forest performs best at 65.5%, followed by the decision tree at 64.0%, logistic regression at 62.2%, and support vector machines at 61.7%, respectively.

This work paves ways for future research in user churn prediction on online language platforms.

References

Ahmad, Abdelrahim Kasem, Assef Jafar, and Kadan Aljoumaa. 2019. Customer churn prediction in telecom using machine learning in big data platform. *Journal of Big Data*, 6(1):28.

Almana, Amal M, Mehmet Sabih Aksoy, and Rasheed Alzahrani. 2014. A survey on data mining techniques in customer churn analysis for telecom industry. *International Journal of Engineering Research and Applications*, 4(5):165–171.

Bernard, Simon, Laurent Heutte, and Sébastien Adam. 2009. Influence of hyperparameters on random forest accuracy. In *International Workshop on Multiple Classifier Systems*, pages 171–180, Springer.

Bisong, Ekaba. 2019. Google colaboratory. In *Building Machine Learning and Deep Learning Models on Google Cloud Platform*. Springer, pages 59–64.

Boser, Bernhard E, Isabelle M Guyon, and Vladimir N Vapnik. 1992. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152.

Brânduşoiu, Ionuţ, Gavril Toderean, and Horia Beleiu. 2016. Methods for churn prediction in the pre-paid mobile telecommunications industry. In *2016 International conference on communications (COMM)*, pages 97–100, IEEE.

Chandrayan, Pramod. 2019. Logistic regression for dummies: A detailed explanation.

Dreiseitl, Stephan and Lucila Ohno-Machado. 2002. Logistic regression and artificial neural network classification models: a methodology review. *Journal of biomedical informatics*, 35(5-6):352–359.

Farquad, Mohammed Abdul Haque, Vadlamani Ravi, and S Bapi Raju. 2014. Churn prediction using comprehensible support vector machine: An analytical crm application. *Applied Soft Computing*, 19:31–40.

Gallo, Amy. 2014. The value of keeping the right customers. *Harvard business review*, 29:2014.

Gaspar, Paulo, Jaime Carbonell, and José Luís Oliveira. 2012. On the parameter optimization of support vector machines for binary classification. *Journal of integrative bioinformatics*, 9(3):33–43.

Hassouna, Mohammed, Ali Tarhini, Tariq Elyas, and Mohammad Saeed AbouTrab. 2016. Customer churn in mobile markets a comparison of techniques. *arXiv preprint arXiv:1607.07792*.

Hsu, Chih-Wei and Chih-Jen Lin. 2002. A comparison of methods for multiclass support vector machines. *IEEE transactions on Neural Networks*, 13(2):415–425.

Hung, Shin-Yuan, David C Yen, and Hsiu-Yu Wang. 2006. Applying data mining to telecom churn management. *Expert Systems with Applications*, 31(3):515–524.

McKinney, Wes. 2012. *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython*. " O'Reilly Media, Inc.".

Min, Jae H and Chulwoo Jeong. 2009. A binary classification method for bankruptcy prediction. *Expert Systems with Applications*, 36(3):5256–5263.

Neslin, Scott A, Sunil Gupta, Wagner Kamakura, Junxiang Lu, and Charlotte H Mason. 2006. Defection detection: Measuring and understanding the predictive accuracy of customer churn models. *Journal of marketing research*, 43(2):204–211.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830.

Sanner, Michel F et al. 1999. Python: a programming language for software integration and development. *J Mol Graph Model*, 17(1):57–61.

Settles, Burr, Chris Brust, Erin Gustafson, Masato Hagiwara, and Nitin Madnani. 2018. Second language acquisition modeling. In *Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications*, pages 56–65.

Singh, Seema. 2018. Understanding the bias-variance trade-off. *Towards Data Science*.

Tomoschuk, Brendan and Jarrett Lovelett. 2018. A memory-sensitive classification model of errors in early second language learning. In *Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications*, pages 231–239.

Tosi, Sandro. 2009. *Matplotlib for Python developers*. Packt Publishing Ltd.

Tsang, Ivor W, James T Kwok, and Pak-Ming Cheung. 2005. Core vector machines: Fast svm training on very large data sets. *Journal of Machine Learning Research*, 6(Apr):363–392.

Vafeiadis, Thanasis, Konstantinos I Diamantaras, George Sarigiannidis, and K Ch Chatzisavvas. 2015. A comparison of machine learning techniques for customer churn prediction. *Simulation*

Modelling Practice and Theory, 55:1–9.

Walt, Stéfan van der, S Chris Colbert, and Gael Varoquaux. 2011. The numpy array: a structure for efficient numerical computation. *Computing in science & engineering*, 13(2):22–30.

Wei, Chih-Ping and I-Tang Chiu. 2002. Turning telecommunications call details to churn prediction: a data mining approach. *Expert systems with applications*, 23(2):103–112.

Xie, Yaya, Xiu Li, EWT Ngai, and Weiyun Ying. 2009. Customer churn prediction using improved balanced random forests. *Expert Systems with Applications*, 36(3):5445–5449.

Zhao, Yu, Bing Li, Xiu Li, Wenhuang Liu, and Shouju Ren. 2005. Customer churn prediction using improved one-class support vector machine. In *International Conference on Advanced Data Mining and Applications*, pages 300–306, Springer.