The Boozy Brain

Silvia Cazacu
Eindhoven University of Technology
Eindhoven, 5612 AZ, Netherlands
s.cazacu@student.tue.nl

Haoyu Dong
Eindhoven University of Technology
Eindhoven, 5612 AZ, Netherlands
h.dong@student.tue.nl

Pranav Muppirishetty
Eindhoven University of Technology
Eindhoven, 5612 AZ, Netherlands
p.muppirishetty@student.tue.nl

Siebren de Vos
Eindhoven University of Technology
Eindhoven, 5612 AZ, Netherlands
s.n.w.d.vos@student.t

Abstract
This paper presents The Boozy Brain (Picture 1), a smart cocktail-machine that uses a supervised machine learning algorithm and a quickly - prototyped interface to generate an alternative to the regular bartender people usually find at parties. First, we will quickly go through similar projects that represented inspiration for our design, then we will explore the theoretical background and the approach the project took. Lastly, a discussion on the projects' limitations and future improvements will be presented followed by a short project conclusion.

Author Keywords
machine learning; supervised learning; Naïve Bayes classifier; sentiment analysis; robot bartender; cocktail machine;

ACM Classification Keywords
H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous;

Introduction
What makes a great party? Experience tells us that when describing the success or failure of a party, people talk about the atmosphere it generated. It is usually created by using the right combination of music, food and drinks. If any of these three ingredients falls behind, people do not feel good and the whole purpose of the party is then removed. All three factors are based on highly subjective standards and likings, thus our team decided to concentrate on just one: the drinks.

Consequently, we looked into different ways to generate drinks that are highly appreciated by people with the least amount of effort from the bartender. We then performed a literature review where we discovered several projects (references) that use feedback given for cocktail recipes based on people's taste. All these examples had an artificial intelligence component which we had considered to design a cocktail machine 'with a brain' that uses the least amount of effort from people to generate new recipes and supports the bartender in preparing them.

Picture 1: The Boozy Brain

Related works

Philippe Remy [1] uses a supervised learning algorithm to determine which cocktail recipes are most likely to be appreciated by party guests. We used his project as inspiration for tackling uneven data distribution issues that may occur in the training phase.

Similar to P. Remy, Yoni Levine [2] uses Natural Language Processing - NLP to analyse which are the most liked coffee recipes based on user reviews focused on taste. In his project, he uses an algorithm that can trace the features on which decisions are based on when rating a coffee. We considered this aspect interesting as a future development for our project.

Scoofy Net [3] is a project that uses IoT to connect devices such as the coffee machine and the air conditioner to the user's smartphone through an app that further memorizes user preferences for better interaction. We believe that a similar feature would positively impact the user experience regarding The Boozy Brain: by generating awareness and synchronization between devices and connectivity between the cocktail machine and Internet people could even taste drinks recommended by their friends from the opposite corner of the world with no extra effort.

Theoretical background

The Boozy Brain uses a supervised learning algorithm based on the Naïve Bayes theorem [4] to determine the cocktail recipes that have a high chance to be positively appreciated by arty guests. The project uses a cocktail recipe database [5] that contains information such as the name, ingredients, volume of each ingredient, average rating and rating count per recipe. This represents the training data for our learning algorithm which in our case can be easily labelled (the user rating), thus the supervised learning choice.

In order to gain an even data distribution for the training phase, the project classified the data as good vs. bad based on the average rating of the recipes used. To classify the data, we needed to use a classifier to assign a class label to the different attributes that describe each recipe. Here we looked into algorithms used for sentiment analysis [6], as the recipe rating represents the opinion people have about certain drinks. Popular applications of sentiment analysis include opinion mining of tweets or by companies to understand their consumers' opinions on their products or services. Here, the ingredients are features of interest instead of words in a tweet.

Such approaches are the Decision Tree, the Support Vector Machine (SVM), Logistic Regression and Naïve Bayes. The team decided to use the Naïve Bayes classifier because of its predictive performance capabilities [7] compared to similar methods, as well as the simplicity and efficiency [8] of the method used to analyse and classify data. Moreover, when compared to artificial neural networks (ANN) the Bayesian algorithm performs better with a smaller training dataset, such us the one used in our project [9].

This classifier is based on an intuitive method of classification that uses the (existing) probabilities of each attribute belonging to each class to make a prediction. It assumes that the effect of a certain variable value on a given class is independent of the values of other variables; the assumption is called class conditional independence [10].

This Bayesian classifier is based on Bayes' Theorem which describes the probability of an event based on prior knowledge of conditions linked to that event's occurrence [4]. Mathematically, it is stated as follows:

$P(A \mid B) = (P(B \mid A) * P(A)) / P(B)$, where A and B are events and $P(B) \neq 0$.

P(A | B) is a conditional probability: the likelihood of event A occurring given that B is true.

P(B | A) is also a conditional probability: the likelihood of event B occurring given that A is true.

P(A) and P(B) are the probabilities of observing A and B independently of each other; this is known as the marginal probability.

When applying this to our dataset, our algorithm looks at two marginal probabilities: the probability of a certain ingredient (say lemon juice) to be present when a drink is randomly selected, and the probability of the drink randomly selected to belong to the class labelled as 'good'. These two are supposedly independent of each other.

The conditional probabilities then emerge: the probability that lemon juice (or previously analysed ingredient) is present in the randomly selected drink labelled as 'good' and lastly, the probability that we have selected a 'good' given that lemon juice is present in the recipe.

At the moment, the project uses this algorithm for combinations of a maximum of 3 ingredients with an accuracy of 69.5%.

Concept overview

To determine whether a cocktail combination generated by the system to be considered likely to be positive or negative, we used the Sentiment Analysis approach. Sentiment analysis is the automated process of understanding an opinion about a given subject from written or spoken language [6].

Algorithm

DATASET GENERATION

The Boozy Brain uses a supervised learning algorithm based on the Naïve Bayes theorem [4]. The data set for training and testing is obtained from a popular cocktail recipe website [5]; using the Beautiful soup library in python [11]. The extracted information including names of the cocktails, ingredients, rating value and rating count are stored into the file "recipes".

Based on the dataset generated, the mean of the rating value can be calculated, which in our case is 3.37. Depending on whether the rating value of the recipe was higher or lower than the mean, the recipe was classified as positive or negative respectively. Each class was stored in one file.

Next, we used sentiment analysis to extract the positive and negative features from the two classes [6]. Take the positive class as the example: each recipe in the positive classed has been broken into words, which in our case are positive ingredients, to form the positive features [6]. Each word or ingredient has two frequency counts, one is the frequency among all the positive ingredients and the other one is the frequency it consists of a positive combination. As mentioned in theoretical background, the probability of the combinations with specific ingredients to be positive could be derived.

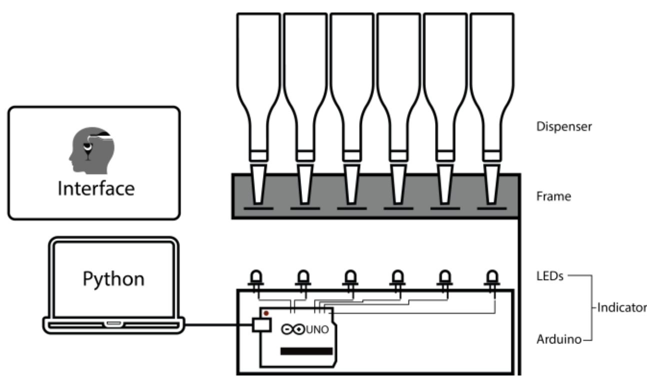
¾ of the features were used to train the Naive Bayes Classifier and ¼ of them were used to test the classifier and calculate the accuracy of the algorithm.

IMPLEMENTATION

After training and testing the algorithm, six chosen ingredients were input into the algorithm. Using the itertool library, the possible combinations can be generated. For the initial implementation, we limited the machine's capability to mix 3 ingredients. After user selecting the desired cocktail combination, the trained algorithm can determine the likelihood of this combination to be positively appreciated by people.

Prototype

In order to provide more interaction between the algorithm and the audience, a physical prototype was built. This prototype consists of four main parts: frame, dispenser, indicator and interface (Picture 2).



Picture 2 – The prototype

The frame was laser cut with the material of MDF, to provide stable support to the dispensers. Then we used wood varnish to give the frame a dark wood texture.

Dispenser is installed on the frame, which can give specific amount (25mL) of ingredients when pressed. The bottles of the dispenser can be removed easily in order to change different ingredients.

The indicator is integrated in a box, which is located at the bottom part of the prototype. It includes six LEDs and an Arduino Uno board. Each LED indicates one ingredient. When a cocktail combination is selected, the LEDs of its composing ingredients will light up.

The interface is designed as a simple series of intelligent dialogue windows. First, it gives the ingredients for the users to choose. After the primary ingredient are chosen, it offers possible combinations with this ingredient. Based on what combination the user chooses, the interface shows the likelihood of being good or bad cocktail. If the chosen combination tends to be bad, the user has another chance to choose.

Discussion and group reflection

The way the algorithm for this project was set up does not allow for a detailed estimation of the rating of newly created mixes. When the mix is selected there is only an estimation with an accuracy of 69.75% whether it is categorised as a positive or negative based on the training data. It would be valuable to the user to know what the estimated rating of the mix would be. This could be achieved by separating the cocktails from the dataset into more than only a ''good'' and ''bad'' category. For example, separating them into multiple categories. This does require an equal amount of mixes in each of the categories if the same algorithm is being used.

The prototype only enabled the user to mix between 6 different ingredients. Thus not fully exploiting the results of the algorithm because this can calculate all possible combinations of ingredients, not only 6.

Future improvements

As mentioned in the discussion, the algorithm calculated the likelihood of a mix being good or bad based on XX ingredients, while only using 6 ingredients in the prototype. This algorithm could easily be connected to a larger system with more tastes.

When you look around in most bars you see a variety of more than 30 different bottles hanging around. Now imagine if each of these bottles had a small wireless led light attached to the bottom of the bottle. Thes leds can be connected to an app or touchscreen standing at the bar. Visitors could than request a drink using the screen and the bartender could easily see which ingredients he would have to use for the cocktail. With using these screens, the process of ordering a drink could be sped up, there will be no more cases of misunderstanding the order on the side of the bartender while still keeping the human touch when receiving the drink. Thus, creating a collaboration between the customer, bartender and intelligent system.

As mentioned in the related work, when combined with the Scoofy concept [3] it is possible to remember personal preferences and get your preferred drink all around the world. This also enables mapping the taste preference from all around the world.

Conclusion

In this project, we aim at exploring the implementation of AI technology into determining what are the cocktail recipes that are more likely to be welcomed by users. Based on the related work and theoretical background research, we used Naive Bayes Theory to discriminate cocktails by considering all the ingredients together as combination. In order to have a functional prototype, we focused on six ingredients and built up an interface and a physical prototype. In summary, we designed the Boozy Brain that could contribute to the beverage industry and replace bartenders. Nevertheless, there is enough space for improvement in the future.

Acknowledgement

We appreciate the instruction and help from Professor Emilia Barakova and Professor Mathias Funk.

References

[1] Philippe Remy. 2016. Create the best cocktails with Deep Learning. Retrieved December 04, 2018 from http://philipperemy.github.io/cocktails/

[2] Levine Y. 2018. The Tale Of Coffee Snobbery and Machine Learning. Retrieved December 09, 2018 from https://towardsdatascience.com/https-medium-com-yoni-levine-coffee-snobbery-45ed534136e5

[3] Ciklum's IoT team. 2016. The Cocktail Mixer Using IoT and Machine Learning Tools. Retrieved December 02, 2018 from https://www.ciklum.com/blog/case-study-cocktail-mixer-using-iot-machine-learning-tools/

[4] Friedman, N., Geiger, D., & Goldszmidt, M. (1997). Bayesian network classifiers. Machine learning, 29(2-3), 131-163.

[5] 1001 Cocktails. Retrieved December 08, 2018 from 1001cocktails.com

[6] Al-Amrani, Y., Lazaar, M., & Elkadiri, K. E. (2017, March). Sentiment Analysis using supervised classification algorithms. In Proceedings of the 2nd

International Conference on Big Data, Cloud and Applications (p. 61). ACM.

[7] Duda, R. O. & P. E. Hart (1973). Pattern Classification and Scene Analysis. New York: John Wiley & Sons.

[8] Melville, Prem, Wojciech Gryc, and Richard D. Lawrence. "SENTIMENT ANALYSIS OF BLOGS BY COMBINING LEXICAL KNOWLEDGE WITH TEXT CLASSIFICATION." Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, Pages 1275-1284, 2009

[9] Forman G., Cohen I. (2004) Learning from Little: Comparison of Classifiers Given Little Training. In: Boulicaut JF., Esposito F., Giannotti F., Pedreschi D. (eds) Knowledge Discovery in Databases: PKDD 2004. PKDD 2004. Lecture Notes in Computer Science, vol 3202. Springer, Berlin, Heidelberg

[10] Varsha Sahayak, Vijaya Shete, Apashabi Pathan. "SENTIMENT ANALYSIS ON TWITTER DATA." International Journal of Innovative Research in Advanced Engineering (IJIRAE), Vol.2, Issue 1, Pages 178-183, January 2015.

[11] Beautiful Soup Library. Retrieved December 08, 2018 from https://www.crummy.com/software/BeautifulSoup/

Appendix

The code can be found on the following external link:

https://github.com/Galactus/BoozyBrain

The video about the project can be found on the following external link:

https://youtu.be/ucPVKxOZiSw