Ffreedom app
Lead Scoring Model

Shaurya Bhardwaj

Aim

Higher conversions

Data Identification

To identify relevant data whose collection will help increase conversions by implementing a model to predict/estimate lead scores.

For this we need data that will not only help us predict the lead score, but also capture which category and subcategories the user is interested in and hence more likely to invest in.

This will enable us to take specific actions for different subgroups which are more suited and applicable to them.

The variable indicative of lead score will be the users expenditure on the app.

User info can be divided into 2 segments

Profile

Gauges the user's profile or long term character in the following subfields:

1. Financial status
   a. Mobile Details (OS,brand,model,RAM)
2. Demographic status
   a. City, State
3. Curiosity/Previous Learning
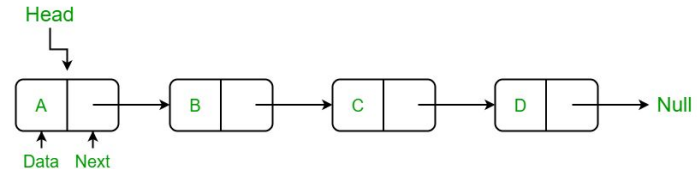   a. Courses completed, Shorts watched

Behavior

Captures the user's (relatively) short term interaction with the application

We can think of this as a time stamped ordered list of the interactions that a user has with the app

This data can be used to extract essential features that will be used in the predictive model

Behavioural data/Interaction

Stored as linked lists of stamps of video watched, pageview, search queried etc

Behavioural Data can be used to compute crucial input statistics like number of sessions, average time per session, course videos/shorts watched (grouped by category) etc

This method will provide us with flexibility in testing and deploying features in the future:

1. Feature engineering
2. Model selection
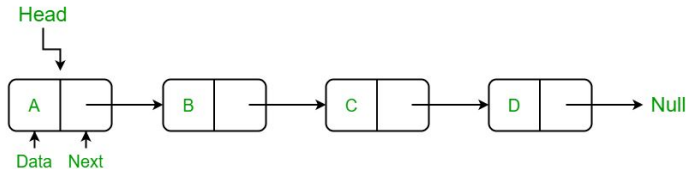
video-stamp

```
{
    "date-time" : "31-12-2024 22:43:33",
    "video-type" : "trailer/shorts/mentor",
    "video-topic" : "business/finance/farming",
    "watch-time" : "03:13",
    "any-other-relevant-detail"
}
```
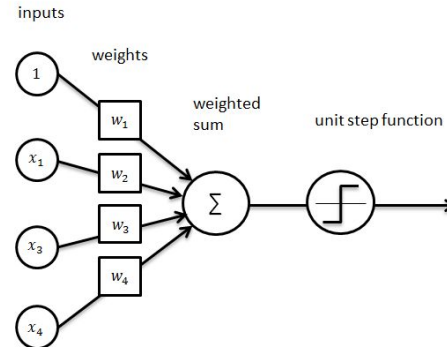
Lead Scoring

The lead scoring models assign points to behaviours and their sum is the lead score of the user. This is called **point allocation**.

We need to perform point allocation for categorical data for example convert financial data like mobile phone details like os,ram etc into a numerical representative form

After point allocation is performed, we need to perform **weighting strategies** to aggregate the data

Behavioral	Points
Form - Fills Out Contact Form	150 points
Tradeshow – Booth visitor	25 points
Event – Attends company-hosted event	50 points
Webinar In House - Attended	50 Points
Webinar Media Sponsored – Attended	50 points
Form - Fills Out Webinar Registration	50 points
Form - Fills Out Content	25 points
Email - Clicks Link	10 points
Web - Visits Case Study Web Pages	15 points
Web - Visits Blog Web Pages	10 points
Web - Visits Multiple Web Pages	15 points

Point Allocation

Based on analytics of historical data we can calculate the financial score, regional score, and curiosity score for each user

These can be scores that are directly proportional to the expenditure done by the specific groups in the app

Statistical measures don't need to be allocated points as they are already in numerical form

The scale of the scores does not matter because:

1. Only relative scores matter
2. A weighting strategy is used
3. Data will be normalised before computing lead scores

| Behavioral	Points
Form - Fills Out Contact Form	150 points
Tradeshow – Booth visitor	25 points
Event – Attends company-hosted event	50 points
Webinar In House - Attended	50 Points
Webinar Media Sponsored – Attended	50 points
Form - Fills Out Webinar Registration	50 points
Form - Fills Out Content	25 points
Email - Clicks Link	10 points
Web - Visits Case Study Web Pages	15 points
Web - Visits Blog Web Pages	10 points
Web - Visits Multiple Web Pages	15 points

Weighting strategy

Instead of a manual weighting strategy, an ML based (learnt) weighting mechanism is proposed.

After point allocation, a user will be represented by a vector which we pass to a deep neural network. The model will be trained in a supervised way to predict the users spending on the app based on his profile and behaviour.

The Multi-layered perceptron will be able to learn the best weights for lead score prediction

The model will learn complex non-linear patterns that are invisible to the human eye.
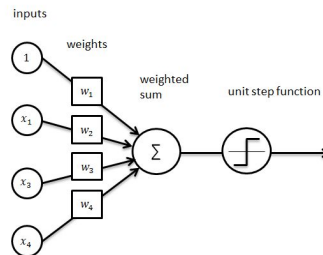
Segmentation and Prioritization

Segmentation and prioritization can be best performed after understanding the business (costs, investments and sales) deeply. However, I will present my understanding.

Lead prioritization should be done based on funds and scope of marketing programs. Broadly speaking, I identify 3 section of leads:

1. Top customers(**highly** likely): These customers have (probably) already made course/subscriptions purchases and our focus should be on retaining them. Investment in free(limited) mentor sessions can be provided to keep them geared towards learning. Also, since they are the main users of the application their valuable feedback must be taken to improve the app. Personalised email updates.
2. Engaging non customers (**moderately** likely): customers who are engaging actively with the free content of the application but are hesitant of investing. Here we can launch offers to "break the ice" i.e. discounted mentor session. Marketing programs should focus on showing these success stories and relevant shorts.
3. Non-engaging (**least** likely): These customers have accounts but are highly unlikely to convert. Even people who do not have an account can fall into this. General ad-campaigns, that increase overall awareness and effectiveness of the application. Referral bonuses should be introduced for customers who bring more users on to the application.

Evaluation and Improvement

Evaluation

1. Hypothesis tests : To test if our mechanism is actually increasing conversion and with what certainty. (ROI with and without using the proposed system)
2. General train-test metrics like accuracy, precision, recall, F1-score etc.

Improvements

1. Feature Engineering
2. Dataset constantly increasing as users are making purchases. So model will keep improving.
3. New models like RNNs or similar time-series models that leverage the entire data and are independent of feature engineering.
4. GenAI