În cele ce urmează, voi descrie procesul meu, de la gândire până la punere în practică(pe cât posibil)  
  
Mai întâi, pentru a aborda problema aceasta, a trebuit să mă uit la date. Am analizat informațiile cu care lucrez. Trebuie să am în vedere exact ce am la dispoziție. După care, am trecut la partea teoretică.

Așadar, ce ar interesa o companie de asigurări? Până la urmă, eu nu trebuie să furnizez date, ci informații. Trebuie să le prelucrez și să le prezint astfel încât să fie folositoare. Deci, din câte știu la momentul actual, interesul principal al companiilor de asigurări este maximizarea profiturilor, adică, investiții sigure, cu risc scăzut. Din abordarea aceasta, pot începe algoritmul, în mare. Este absurd să încep să mă gândesc la programare propriu-zisă până când nu știu ce vreau să facă programul meu. Este deosebit de important să știu ceea ce vreau să fac prin scrierea acelui cod. Așa pot începe acea parte a muncii cu obiective clare. Muncă fără rost este muncă pierdută.

Eu am decis să creez ceva modular, versatil. Pentru a demonstra exact la ce mă refer, am folosit o schemă bloc. Această schemă bloc are blocurile individuale dar și segmente, după cum le-am denumit. Deci, pornim de la primul segment, sau segmentul 0. Acest segment poate avea mai multe blocuri, dar acesta este cel pe baza căruia se face clasificarea.

Alegerea Sectorului

Ridicat

Mediu

Scăzut

Cât risc preliminar?

Schema bloc de mai sus reprezintă segmentul 0, pentru prima soluție. Mai întâi, se caută sectorul. Astfel încât, firma își poate alege în ce domeniu să investească. După care, riscul preliminar. Acest risc preliminar, după cum l-am denumit, se referă la o medie a riscului per total. Spre exemplu, în sectorul educației, riscul, de obicei este scăzut. Există și alte activități sau firme care pot avea un risc ridicat, dar, să zicem, cu o frecvență foarte redusă. Așadar, poate apărea un risc colosal pentru o firmă, dar care se va întâmpla, sau nu, doar o singură dată, în viața companiei. Aceasta ar intra la categoria de risc scăzut. Prin alegerea sectorului prima dată, firma de asigurări își poate face o idee inițială. După aceea, se sortează în funcție de risc. Aceasta fiind prima soluție(în totalitatea conceptului), are ca avantaj că poate fi modificată relativ ușor și în funcție de cerințe specifice. Mai ales, este simplu, adică, simplu de înțeles, simplu de implementat și care permite ajustări foarte multe, având în vedere că este în stadiul de prototipare. Un dezavantaj apare atunci când se dorește o clasificare sau sortarea din ce în ce mai detaliată. Acest fapt va crește numărul de segmente și implicit va crește complexitatea programului în sine, alături de resursele necesare pentru procesare.

Ar trebui să menționez faptul că se poate alege, pentru segmentul 0, riscul prima dată. Sau un alt criteriu, cum ar fi locația unei firme. Sau se poate începe cu un alt segment diferit. Pentru a determina eficiența exactă a soluției, trebuie testată, pusă în practică. Chiar dacă pare genial pe hârtie, rezultatele se obțin pe teren.

Următorul segment, adică segmentul 1, poate fi legat de contribuția la PIB a firmei respectivă. Deci, după ce se alege riscul preliminar, se alege valoare monetară sau economică, astfel încât să fie clar cu ce sume se lucrează. Pentru un exemplu mai concret, următoarele date se referă la România și sumele aferente sunt în euro.

>8mil(entități mijlocii și mari)

<8mil(entități mici)

<700k(microentități)

Cât de mare e contribuția la PIB?

Conform legii din România(OMFP 1802/2014), firmele din România sunt clasate în microentități, entități mici și entități mijlocii și mari. Există mai multe criterii de clasificare, în afară de cifra de afaceri, ce a fost utilizată pentru segmentul 1, respectiv active totale și număr mediu de salariați. Trebuie precizat faptul că mai există grupuri mici, mijlocii și mari(constituite din societățile-mamă și filiale) a căror cifră de afaceri netă este sub sau peste nivelul de 48 de milioane de euro. Dar, pentru un exemplu practic, sumele menționate anterior sunt suficiente. Totuși, mai este nevoie de o sortare la acest nivel pentru a determina exact cât de mari sunt entitățile comerciale. Apoi, se poate continua cu alte segmente care se referă la locație, angajați, etichete specifice de afaceri etc.

Prototipul 1, adică primul algoritm pus în practică folosește segmentele prezentate anterior. Această etichetare/sortare este rudimentară, dar, suficientă pentru ceva estimativ, de prim-plan. La acest moment, abilitatea practică necesară pentru scrierea codului este în afara expertizei mele. Totuși, codul, care este în C++, ar trebui să conțină: structuri care să aibă etichetele și companiile și apoi funcții pentru recunoașterea companiilor.

#include <iostream>

#include <string>

#include <vector>

#include <algorithm>

using namespace std;

struct Company {

string name;

string description;

vector<string> tags;

};

struct Label {

string name;

vector<string> keywords;

};

int main() {

vector<Label> taxonomy = {

{"Health Insurance", {"health", "medical", "hospital"}},

{"Auto Insurance", {"car", "auto", "vehicle"}},

{"Life Insurance", {"life", "mortgage", "beneficiary"}}

};

For(...){

If(...){

...

}

}

Cout<<company.name <<endl;

Cout<<`` ``<<label;

return 0;

}

Există multe funcții și modalități de rezolvare. Unele sunt mai eficiente decât altele. Probabil, Python ar fi o alegere mai bună pentru a lucra cu baze de date, decât C++.

După cum s-a menționat anterior, un avantaj îl reprezintă simplitatea algoritmului și modularitatea sa. După cum se poate vedea în cod, este destul de banal și ușor de înțeles, dar, atunci când se dorește o sortare mai amănunțită, complexitatea sa va crește și, implicit, resursele necesare pentru computație. Eficiența codului în sine este greu de stabilit, având în vedere că este incomplet, dar, mai mult, acesta trebuie testat și practic, de către firmele de asigurare. Nu se poate obține un rezultat mai clar de atât. Desigur, este necesară și o testare internă sau o testare alta decât cea din practică, pentru a observa și alte aspecte precum limitări tehnice sau greșeli în design, astfel, să poată fi depistate înainte să fie înmânată clientului.