Correcting addresses written in any language

Author: Druţă Georgiana
Coordinator: Profesor Dr. Iftene Adrian

June, 2022

Universitatea "Alexandru-Ioan Cuza" din Iaşi
Facultatea de Informatică

Summary

1. Motivation
2. Conventions in writing the address
3. Who decides if an address is correct?
4. Challenges
5. Solution
6. Demo
7. Conclusions

Motivation

Motivation

The need to verify the address of each online order (user side and service provider).

But what happens when you incorrectly fill out a delivery form in the United States where you are not called to verify the address and your package is left at the door?

Conventions in writing the address

Conventions in writing the address

The addresses will follow the following form: **"street line, zip code, state, city, country"**.

Who decides if an address is correct?

Correctness of an address

A person who handles the delivery of orders on a daily basis can easily figure out the correct address.

For example, delivery will still be made if the addresses are:

- **Starda** General Henri Mathias Berthelot Nr. 16, 700259, Iaşi, Iaşi, România (spelling mistake)
- **Str.** General Henri Mathias Berthelot Nr. 16, 700259, Iaşi, Iaşi, România (variation mistake)
- Str. General Henri Mathias Berthelot Nr. 16, 700259, Iaşi, **-**, România (omitting a field)
- Str. General Henri Mathias Berthelot Nr. 16, 700259, Iaşi, Iaşi, România, **USA** (conflict)
- **Facultatea de Informatică**, Iaşi, România (specifying a point of interest)

Challenges

Challenges

- Best possible algorithm accuracy **80-90%**
- An algorithm as **fast** as possible
- Generalization of the solution for **worldwide** addresses

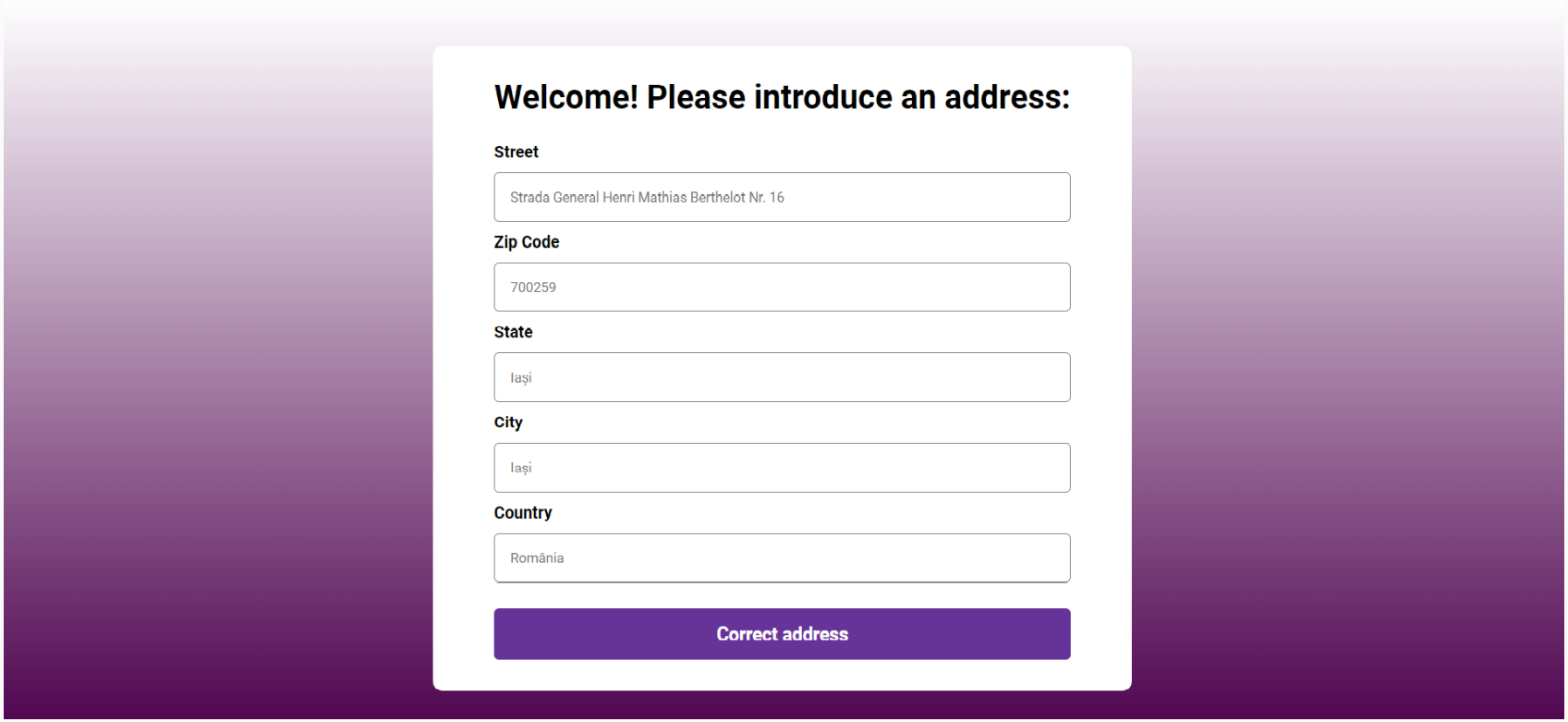

Figura 1: Form page

Solution

Detailing the solution

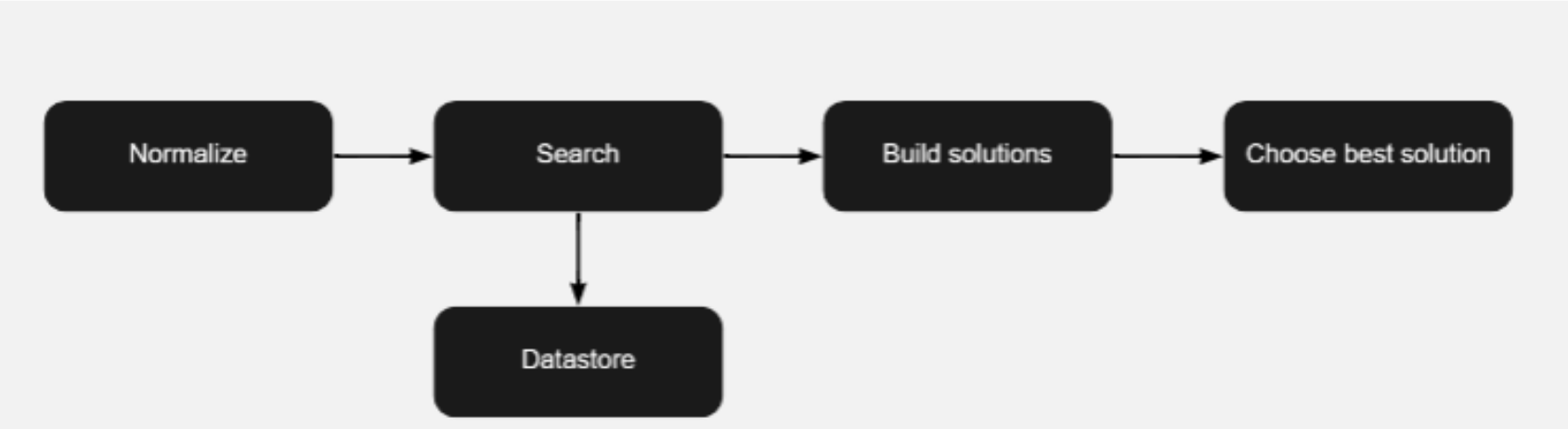

Figura 2: Steps

The given address:

- Street line: Strada General Henri Mathias Berthelot Nr. 16
- Zip code: 700259
- State:
- City: Iaşi
- Country:

Normalization of the address

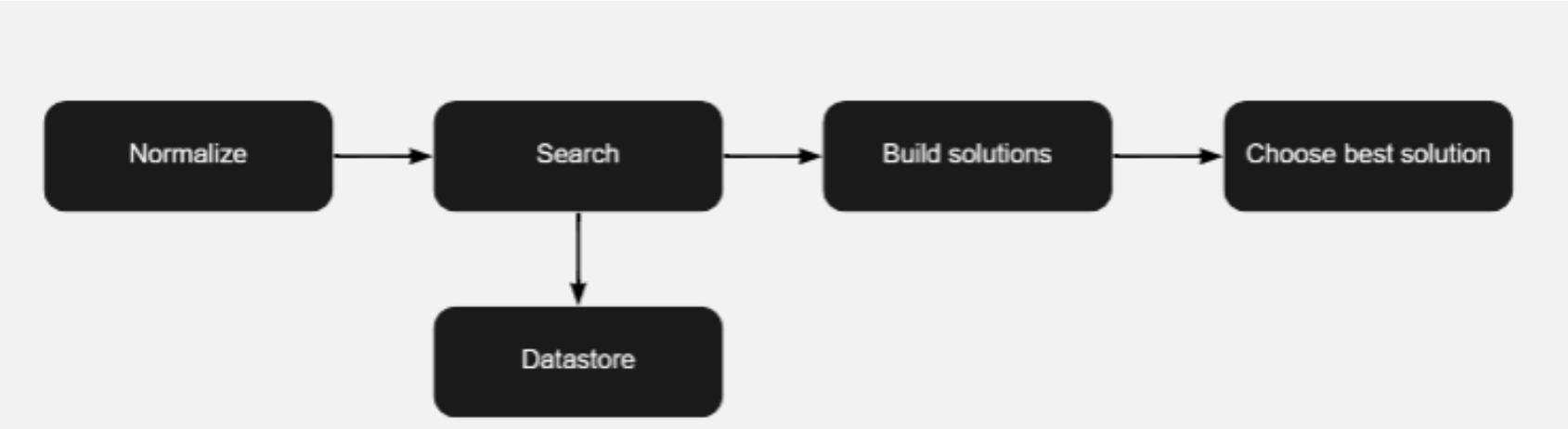

Figura 3: Steps

Normalized address:

- Street line: strada general henri mathias berthelot nr 16
- Zip code: 700259
- State:
- City: iasi
- Country:

Normalization of the address

Normalization, bringing it to a canonical form, involves translating addresses that do not contain **only ASCII characters** and other classic operations such as: **removing multiple white-spaces and special characters and writing in lower case**.

```
input = input.toLowerCase();
input = Normalizer.normalize(input, Normalizer.Form.NFD).replaceAll( regex: "\\p{InCombiningDiacriticalMarks}+", EMPTY_STRING);
newList.add(input
        .replaceAll( regex: "(\\D)\\.(\\d)", replacement: "$1 $2")
        .replaceAll( regex: "[!@€£¢$¥%^*\"`~><()\\-.=_;,\\\\+?{}\\[\\]:|\\s]+", ONE_WHITESPACE)
        .replaceAll( regex: "[-\\-]+", ONE_WHITESPACE)
        .replaceAll( regex: "#+", ONE_WHITESPACE)
        .replaceAll( regex: "/+", ONE_WHITESPACE)
        .replaceAll( regex: "&+", replacement: " and ")
        .replaceAll( regex: "°+", replacement: "o ")
        .replaceAll( regex: "[`']+", EMPTY_STRING)
        .replaceAll( regex: "[.]+", ONE_WHITESPACE)
        .replaceAll( regex: "\\s+", ONE_WHITESPACE)
        .trim());
```

Figura 4: Normalization of the address

Corrected address

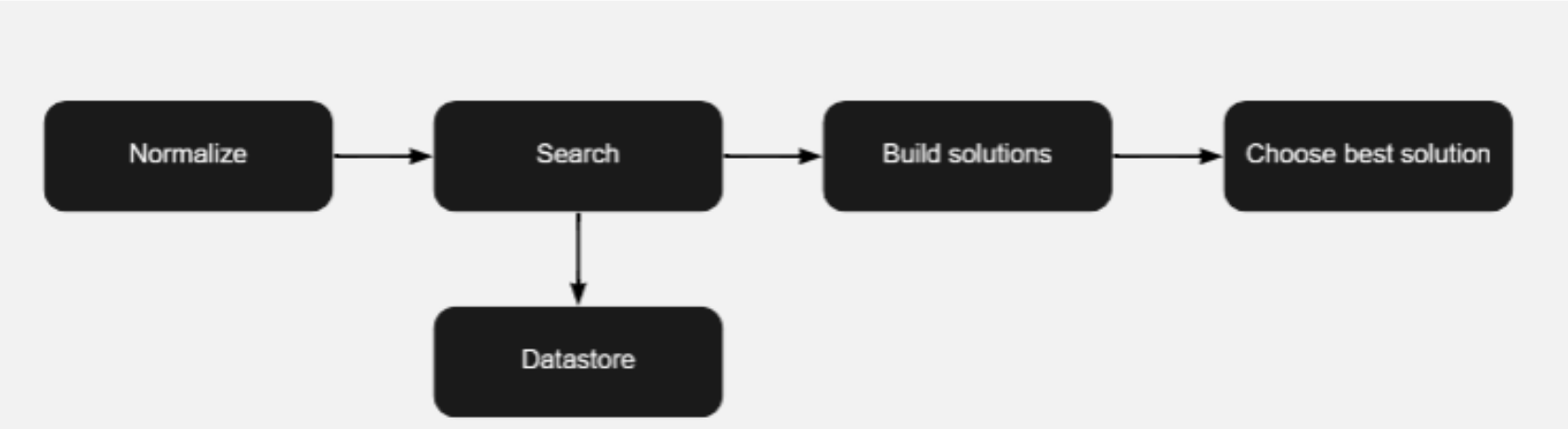

Figura 5: Steps

Corrected address:

- Street line: strada general henri mathias berthelot nr 16
- Zip code: 700259
- State: iasi
- City: iasi
- Country: romania

Administrative chains

Represents the chain of country, state and city, limited to 3 levels (according to the chosen convention).
Examples:

- România -> Iaşi -> Iaşi
- US -> New York -> New York
- Iran -> Teheran Province -> Teheran

Tree collections

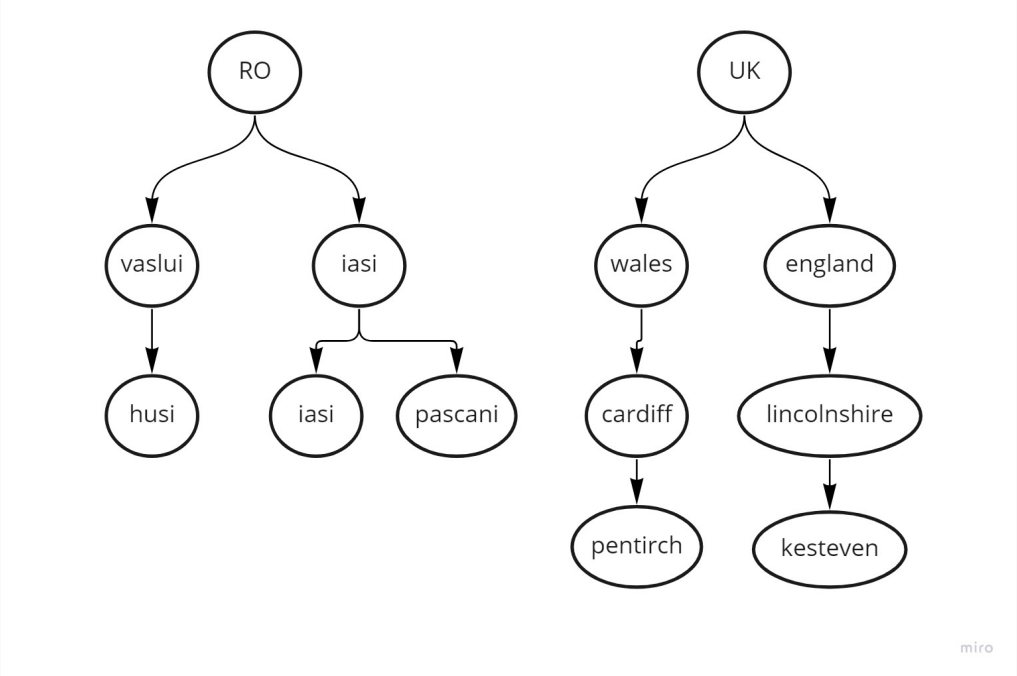

Figura 6: Links between locations

Using the **tree-type data structure**, the links between locations are preserved. This way we will be able to create the most powerful solution.

Database

Database contains a collection of trees for each country, **252 trees**.
Each tree contains X00.000 vertices, where X is a positive number, each node being as detailed as possible.

Searching algorithm

The search will be done in every field, even if the street line and zip code fields are not in the database.
What if a field contains **multiple pieces of information**? The algorithm extracts as much information as possible, starting with the entire value in the field reaching one word at a time.
City: Georgiana New York Druţă

- Georgiana New York Dru - no hit
- New York Dru - no hit
- Georgiana New - no hit
- New York - hit
- Georgiana - no hit
- Dru - no hit

Searching algorithm

One node can have several children. To search for a node we must go through the tree in depth or width.
How much would the search process be streamlined if this were avoided?
What if this is possible in **O(1)**?

Ambiguous addresses searching

How many alternative names can a location have?

- ro | romania | rumänien | rumanien | românia | românia | roumanie | rumanía | rumania | rom
- us | united states | usa | u s | united states of america | united states america | america | vereinigte staaten | états unis | etats unis | estados unidos | u s a

This makes it possible to correct addresses in **multiple languages and for different variations**.

Building solutions

- Extracting information from fields
- Building intermediate solutions
- Sorting valid solutions
- Choosing the solution with the highest trust score

Scoring algorithm

How much confidence can we have in a possible address? How do we know which one is the best?

SCORE = a * 5 + b * 3 - c * (-3) + n, unde

a = 1 dacă valoarea se află în câmpul corect, altfel 0,
b = 1 dacă valoarea din câmpul city este un oraș diferit de județ, altfel 0,
c = 1 dacă valoarea se termină cu "*", altfel 0,
n = lungimea listei ordonate cu informațiile rezultate din adresa dată - indexul valorii în această listă

Figura 7: Calculating score

Address: Str. Sfantul Lazar Nr. 28, 247752, Vâlcea, Voineasa, România.
The first two solutions with the highest score are:

- solution 1: str. sfantul lazar nr 28, 247752, valcea, voineasa, romania -> scorul **25**
- solution 2: str sfantul lazar nr 28, 247752, valcea, ramnicu valcea, romania -> scorul 20

Average precision

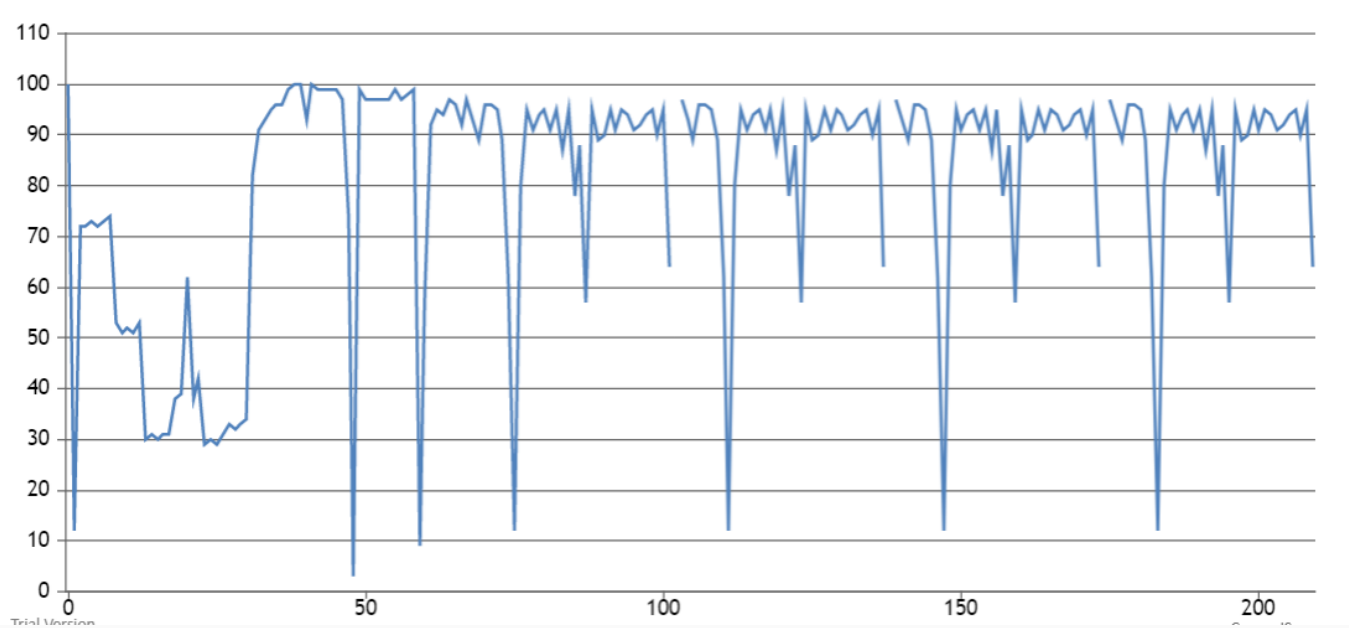

Figura 8: Precision metric for Romania

Country	Average precision	Time
România	89%	7s
Japonia	78%	7s
Germania	75%	7s

Tabela 1: Comparison between accuracy and address correction time

Demo

Conclusions

Conclusions

A web application has been created that allows the user to enter an address that contains at least one completed field, and then view a list of the three best possible corrected addresses.

Possible directions for development:

- Creating the entire database
- Retrieving data from the database
- Integration with other platforms

References i