**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**Khoa Công nghệ Thông tin**

**DỰ ĐOÁN BẤT ĐỘNG SẢN TP.HCM**

*- Tìm hiều về phương pháp học máy SVM-R.*

*- Biết được cách hoạt động của mô hình SVM-R.*

*- Tìm hiểu cách đánh giá mô hình, lựa chọn thông số phù hợp có thể áp dụng cho nhiều dạng bài toán khác nhau.*

*- Hiểu được cách áp dụng SVM để có thể huấn luyện máy học cho bài toán hồi quy để dự đoán giá nhà bất động sản. Từ đó, rút ra những nhận xét dựa trên kết quả thực nghiệm mang lại.*

C:\Users\tdqua_000\Dropbox\SS-Slides\DeCuong-CDIO\Template CDIO v4.2\Templates\Hinh anh\LogoTruong.png

Data Sciences

Khoa Công nghệ thông tin

Đại học Khoa học tự nhiên TP HCM

1. THÔNG TIN SINH VIÊN

|  |  |
| --- | --- |
| Họ và Tên | MSSV |
| Hoàng Nhật Sương | 1412470 |

1. BÀI TOÁN ĐẶT RA

* **Làm sao biết được giá trị căn nhà mình muốn rao bán?**

**Real estate forecast** hay còn gọi là dự đoán bất động sản. Đó là mô hình được đặt ra nhằm đưa ra hàm dự đoán giá cả của một căn hộ bất động sản nằm trong phạm vi thành phố Hồ Chí Minh dựa trên các đặc tính của căn hộ đó.

**Ý nghĩa thực tế của mô hình:** *có rất nhiều người băn khoăn không biết với ngôi nhà mà mình đang sở hữu mình có thể rao bán trong khoảng giá như thế nào là hợp lý, bài toán trên sẽ giúp họ giải quyết vấn đề đó, bằng cung cấp thông tin về ngôi nhà mà họ đang sở hữu, hệ thống sẽ phân tích và đưa ra mức giá dự đoán.*

1. THỰC NGHIỆM

* **PHÂN TÍCH CÁC YẾU TỐ ẢNH HƯỞNG TỚI GIÁ BĐS**

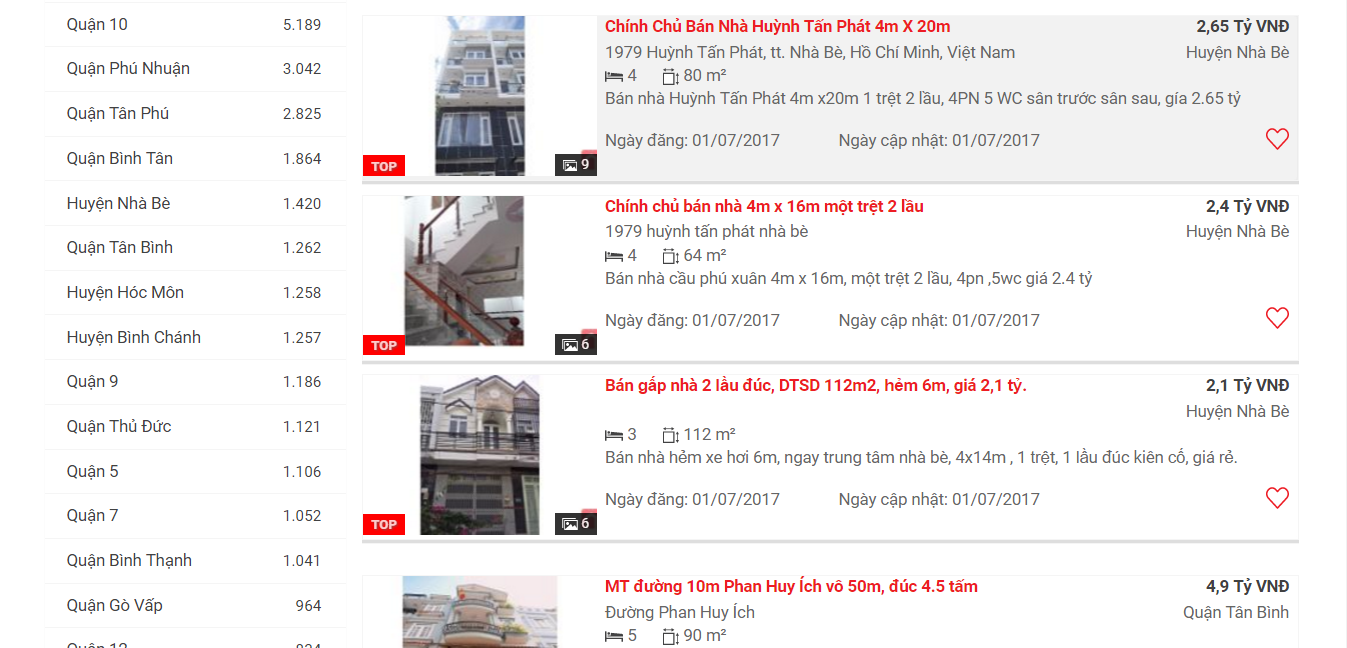
Sau khi phân tích và nghiên cứu từ trang web **muabannhadat.vn**, mình đưa ra được 11 thuộc tính chính trong việc ảnh hưởng tới giá nhà đất tại một thời điểm trong hiện tại.

**Quận, Phường, Toạ độ, Loại bán, Diện Tích, Số phòng ngủ, Số phòng tắm, Số tầng, Môi trường xung quanh, Tiện ích**

* **KHAI THÁC DỮ LIỆU TỪ HTML**

**Nguồn: http://www.muabannhadat.vn/**

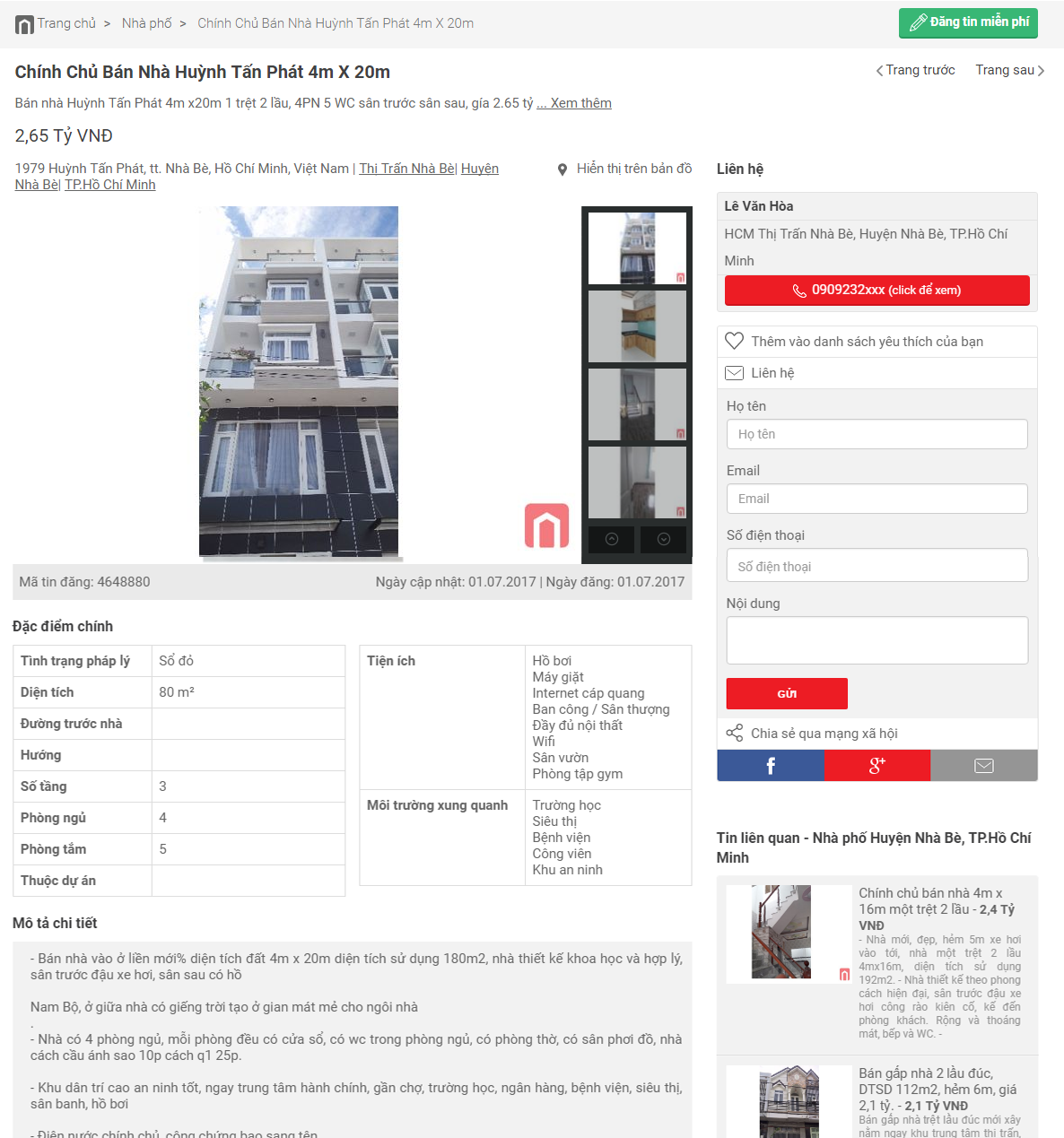
**Khai thác các link chứa các bài đăng bán:**



Hình 1 List các bài đăng bán

Sử dụng thư viện BeautifulSoup để parse html và thu thập dữ liệu.

**Khai thác các thuộc tính của căn nhà:**

****

Hình 2 Bài post đăng bán

Cũng bằng phương pháp parse html tree, mình lần lượt khai thác các thuộc tính mà đã được nêu ra trên bước 1.

Trong lúc khai thác, mình sẽ bỏ qua các bài đăng để trống quá nhiều thuộc tính chỉ chừa lại các bài đăng có trên 50% thuộc tính được điền.

Sau khi khai thác, mình có được file dữ liệu thô như sau:



Hình 3 Dữ liệu thô sau khi khai thác

**Kết quả:** ***2056 bộ dữ liệu, 12 thuộc tính. Từ đó mình chia ra 3 tập dữ liệu Train, Validate, Test với tỉ lệ 6 – 2 – 2.***

* **Tiền xử lý**

Ban đầu mình sẽ dò qua bộ dữ liệu, qua đó mình phát hiện là có rất nhiều các dữ liệu sai thực tế.

* **Dữ liệu bị lạc từ các Vùng khác như** ***Vũng Tàu, Huế, Hà Nội***….
* **Dữ liệu bị nhập sai so với quy chuẩn của trang web** (vd: Phường 2 thì bị ghi thành P.2)
* **Dữ liệu có toạ độ ảo**

*([0.0, 0.0], [10.8231, 106.62966] trường hợp này phát sinh do trang web chưa thống kê các tuyến đường nằm trên phường, Quận nào đấy mà cho phép người dùng nhập trực tiếp vào nên API google maps không thể tìm ra được tạo độ đúng mà cho giá trị ảo)*

* **Các Phường theo quy chuẩn trước khi quy hoạch theo quy định mới của chính phủ.**

*Vd: giả sử như Phường Bình Hưng Hoà thuộc Quận Bình Tân sau khi quy hoạch nhưng do sai sót nên có vài căn hộ khi rao bán trên muabannhadat.vn thì vẫn giữ theo chuẩn cũ tức là xã Bình Hưng Hoà thuộc Huyện Bình Chánh cũ nên những sai sót như vậy mình vẫn có thể xử lý cho đúng đắn được, do thế, mình đã so sánh với thống kê của cục thành phố về các phường của các Quận ở file “*[*HCM.xlsx*](HCM.xlsx)*”*

Sau khi xoá đi các bộ dữ liệu sai sót, mình tiến hành điền và thay thế các toạ độ ảo bằng cách gom nhóm các bộ dữ liệu có thuộc tính ***‘Quận’*** và ***‘Phường’*** giống nhau. Từ đó lấy 1 giá trị toạ độ trong nhóm đó điền vào giá trị của bộ dữ liệu bị sai.

Sau khi điền xong các giá trị sai của toạ độ, mình sẽ xoá đi thuộc tính ***‘Phường’***, do theo mình phân tích thì trong mỗi quận sẽ có rất nhiều phường, nếu sau này mình chuẩn hoá về lại dữ liệu số thì thuộc tính ***‘Phường’*** là rời rạc nên khi chuyển qua dạng hotkey sẽ bị trùng nhau giữa các quận với nhau, nên mình sẽ xoá nó đi thay vào đó mình sẽ dùng toạ độ để tránh việc trùng nhau đó.

Sau đó mình sẽ điền các giá trị thiếu bằng cách gom nhóm trong tập train các dữ liệu có cùng ***‘Quận’*** và ***‘Diện Tích’*** sẽ nằm trong khoảng sai số nhất định với dòng cần điền *(vì 2 thuộc tính này luôn luôn đủ),* với các thuộc tính chữ mình sẽ điền bằng giá trị xuất hiện nhiều nhất trong nhóm đó, nếu là giá trị số, mình sẽ điền bằng cách tìm ra độ sai số trung bình của cột giá trị đó trong nhóm đã gom trước, sau đó tính giá trị trung bình của cột giá trị nhóm đó và random một giá trị nằm trong khoảng ước lượng đó để điền vào.

Cuối cùng mình sẽ chuyển toàn bộ các thuộc tính chữ sang dạng số, do ở đây các thuộc tính chữ của mình là các thuộc tính rời rạc nên mình chuyển tất cả về dạng hotkey.

Ở bước tiền xử lý, mình đã quyết định không scale lại vì dữ liệu của mình thu thập theo thứ tự từ 6 tháng trước cho tới thời điểm gần nhất nên xảy ra tình trạng việc phân bố xác suất khác nhau giữa các tháng. Mặt khác do mình xử dụng mô hình học tuyến tính nên việc scale dữ liệu như thế này cũng không giúp nhiều lắm về thời gian trainnning nên mình quyết định bỏ scale dữ liệu.

* **Máy học**

Mình sử dụng mô hình học ***Support Vector Regression*** để huấn luyện trên môi trường hệ điều hành ***windows 10 Education, RAM 4Gb, CPU core I5U.***

***Kết quả:***

Tập train: A screenshot of a computer

Description generated with very high confidence

Tập Validation:

A screenshot of a computer

Description generated with very high confidence

Qua các bộ thử trên mình thấy với gamma bằng 0.02 thì mình sẽ thu được giá trị độ lỗi tốt nhất. Sau đó mình đã thử với các bộ C = [600000000, 800000000, 10000000000]

Thì mình thu được kết quả của tập c = 800000000, gamma = 0.02 tốt nhất và gần đây nhất:

**C = 800000000, gamma = 0.02**

**Train: 0.402043171505**

**Validate: 0.388047291902**

**Test Error: 0.32734016902649421**

1. NHẬN XÉT

* **Về tập dữ liệu:**

**Do bài toán dùng dữ liệu khai thác từ trang web muabannhadat.vn nên giá nhà thu được là giá trước khi bán (có thể sai với giá bán thực tế) nên có nhiều nhiễu. Bị trống nhiều.**

* **Về mô hình máy học SVR:**

**Sau khi thực nghiệm ta có thể nhận thấy sự thay đổi của C và gamma gây ảnh hưởng rất nhiều tới kết quả dự đoán. Khi C quá nhỏ, thì Etrain rất lớn tức là khi đó xảy vấn đề underfitting. Khi C quá lớn thì Etrain hầu như bằng 0 và thời gian training cũng rất lớn, chứng tỏ đã xuất hiện tình trạng overfitting.**

**Tương tự như C, với Gamma khi quá bé cũng xảy ra tình trạng underfitting và khi quá lớn cũng xảy ra overfitting.**

**Nên khi huấn luyện ta cần lựa chọn giá trị c và gamma trung dung để có kết quả tốt nhất.**

**Ngoài ra trong SVR còn có thêm một giá trị có tên gọi là epsilon, giá trị của epsilon sẽ quyết định độ lớn của một khoảng xung quanh hàm dự đoán. Những vector nằm ngoài khoảng đó đều được coi là sai.**