Title of submission to PLOS journal

Yipeng Lai [2] , Lauren Low [1] , Dayana Meza [1] , Emma Scott [1] , Elaine Ye [1] , Yanwan Zhu [1]

1 1 Chapin Way, Northampton, MA 01063
2 1250 Broadway, New York, NY 10001

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur eget porta erat. Morbi consectetur est vel gravida pretium. Suspendisse ut dui eu ante cursus gravida non sed sem. Nullam sapien tellus, commodo id velit id, eleifend volutpat quam. Phasellus mauris velit, dapibus finibus elementum vel, pulvinar non tellus. Nunc pellentesque pretium diam, quis maximus dolor faucibus id. Nunc convallis sodales ante, ut ullamcorper est egestas vitae. Nam sit amet enim ultrices, ultrices elit pulvinar, volutpat risus.

Author summary

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur eget porta erat. Morbi consectetur est vel gravida pretium. Suspendisse ut dui eu ante cursus gravida non sed sem. Nullam sapien tellus, commodo id velit id, eleifend volutpat quam. Phasellus mauris velit, dapibus finibus elementum vel, pulvinar non tellus. Nunc pellentesque pretium diam, quis maximus dolor faucibus id. Nunc convallis sodales ante, ut ullamcorper est egestas vitae. Nam sit amet enim ultrices, ultrices elit pulvinar, volutpat risus.

Text based on plos sample manuscript, see
https://journals.plos.org/ploscompbiol/s/latex

Introduction

Real estate is one of, if not the biggest, investments an individual makes over the course of their lifetime. Traditionally, home valuation has been conducted by professional property appraisers – those licensed and hired to give an opinion of a home's value, based on prices of neighboring homes, property analysis and judgment. In recent years though, there has been an uptick in developing models and algorithmic resources to aid housing price estimates. Companies like Zillow, Trulia, and others have developed machine learning models to automate the job of an appraiser. These organizations take into account housing attributes including property size, number of bedrooms, number of bathrooms, geographic location, state of the economy, among other variables to predict home prices.

StreetEasy, launched in 2006 and acquired by Zillow in 2013, is reshaping the ways people buy, sell and rent property in New York City and New Jersey. StreetEasy's existing predictive tool uses metrics on bedrooms, bathrooms, property size, and geographic features to estimate particular listing prices, similar to Zillow's Zestimate.

One feature the current machine learning algorithm neglects is the descriptive text of given listings. A primary objective of this project is to convert text-based listing descriptions into input variables for the purpose of improving predictive model accuracy. In particular, we want to research whether listing descriptions can be utilized to create meaningful features to improve the accuracy of predicting home prices. Past literature supports leveraging machine learning algorithms with natural language processing to improve prediction accuracy [1,2].

Research Questions and Objectives

Some research questions include:

- Can we use listing attributes to predict home prices?
- Can we convert the description text into meaningful features?
- Can we use text-based features in addition to the existing features to predict home prices?

Many questions already have preliminary answers. In StreetEasy's current model, home prices are predicted using variables such as number of bedrooms, number of bathrooms, property size, time to subway, year the property was built and property neighborhood/geographic coordinates. To predict home prices more accurately, we include the listing description in the model. Using bag-of-words method and word embeddings to extract keywords, our analysis has yielded some key phrases such as "stainless steel," "washer dryer," "windows," "storage," "central park" and "closet space".

In asking and answering these questions our goal is to improve upon StreetEasy's existing machine learning algorithm by combining current models with text-based analysis to predict home prices. By creating new variables from the listing descriptions – binary variables for the term "stainless steel", say – we aim to improve prediction accuracy.

Methods

Data and Variables

The data provided by our project partner includes two datasets on sale listings and amenities. StreetEasy collects information from agents by having them directly enter listing information on the website or through a feed from their brokerage. StreetEasy verifies the listing and property information based on records from New York City's Department of Finance and Department of Buildings.

Our main data set consists of 59,661 sale listings of 52639 unique properties listed and/or sold on Streeteasy in 2019. It has 31 variables including home price, property attributes (e.g., the number of bedrooms, the number of bathrooms, size in square feet), geospatial information (e.g. state, zipcode, latitude, latitude), and a text description of each listing.

(Make a table of variable names and descriptions?)

Data Preprocessing

(More description/plots of EDA here?) Exploratory data analyses show that the distributions of the variables are skewed and 13 of them contain NA values for up to 11242 observations. We performed data cleansing, reduction, and imputation to pre-process our data before running the analyses. First, we select the 95% quantile of

the non-zero values as a reasonable range of the listing prices. We also filter out the outliers with unusually large values for property size and number of bedrooms and bathrooms.

In addition, we use zip codes to fill in an estimate for observations without valid latitude/longitude information and created more geographical variables such as city, county, and state using the R package zipcodeR that pulls information from U.S. Census data. We perform these two steps because of the varying quality of geospatial information in the original data set. First, some of the listings contain four-digit zip codes and erroneous longitude and latitude values such as (0, 0). Secondly, the original location information about areas and neighborhoods was entered manually by agents, so the location variables do not have uniform criteria and may differ in the level of specificity. Although the new geographic coordinates might not have street-level precision, by adding a leading zero to four-digit zip codes and incorporating associated features from the zipcodeR package, we are able to obtain geographic information that is consistent across listings.

In order to prevent data leakage for our machine learning models, we remove the listings that appear more than once in the data set. The duplicate listings are results of agents posting the same listing multiple times with updated information. In order to eliminate duplicates, we first group the listings by `property_id` and keep the ones with the fewest NA values for all variables within each group. Next, we select the rows with the longest listing descriptions and the largest number of bedrooms within each group. Last, we use the distinct() function to keep one unique listing corresponding to each `property_id`.

Because nearly 20% of the observations do not have values for the `size_sqft` variable, we plan to use the `mice` package to impute missing property size values by regression (`norm.predict`). The predictors of the imputation consist of the number of bedrooms, the number of bathrooms, unit type, floor count, city, and state.

Data Analysis Plans

In order to determine whether adding text features as predictors will improve the accuracy of predicting housing price, we first create models without text features and then compare their model evaluation metrics with those with extracted text features. We select Root Mean Squared Log Error (RMSLE) as our evaluation metrics for the linear regression model; the smaller the RMSLE, the more accurate the model is in predicting the outcome. To avoid overfitting and ensure the generalizability of our models, we plan to perform k-fold cross validation on each of our models and take the mean RMSLE as the metrics for comparison across models.

To establish a baseline, we create a null model that only has mean log price as the predictor. We expect all other models we build to have better prediction performances than the null model.

Models without text features. We are working on building a linear regression model using log price as outcome and a decision tree/ random forest model that automatically predicts which categories of price range a listing falls into. For the linear regression model, we plan to use stepwise selection to select the top 5 most significant predictors of housing price, build the model using those variables, conduct 5-fold cross validation, and then compute the mean RMSLE.

Models with Text Features

To convert the listing descriptions into meaningful text-based features, the words are (1) tokenized: the words inside of the listing descriptions are split into word or words (i.e., multiple words tokenzied together are called 'n-grams') known as tokens to create a

one-token-per-row for text analysis; (2) vetted for stop words: removing words that have little to no meaningful information for analysis (e.g., 'the', 'as', 'a', 'of'); (3) stemmed: finding the base or root of words (e.g., we want to categorize 'windows' the same as 'window'), and lastly (4) analyzed for extracting keywords and n-grams. **[add a sentence here about the benefits of including n-grams in model]** N-grams generate words that commonly occur together and improve accuracy of sentiment analysis models [3,4]. This allows us to infer the value from text using regular expressions or models.

Different ways for featurizing text data:

- Define a target feature/tag such as "whether the apartment has been recently renovated" and infer the value from text using regular expressions or models
- Use bag-of-words method by counting the frequency of words observed in each document
- Use pre-trained word embeddings or train word embeddings

Here are two sample references: [5,6].

References

1. Chen P-H, Zafar H, Galperin-Aizenberg M, Cook T. Integrating natural language processing and machine learning algorithms to categorize oncologic response in radiology reports. Journal of digital imaging. Springer; 2018;31: 178–184.

2. Ivanov O, Wolf L, Brecher D, Masek K, Lewis E, Liu S, et al. Improving emergency department esi acuity assignment using machine learning and clinical natural language processing [Internet]. 2020. Available: `http://arxiv.org/abs/2004.05184`

3. Kruczek J, Kruczek P, Kuta M. Are n-gram categories helpful in text classification? In: Krzhizhanovskaya VV, Závodszky G, Lees MH, Dongarra JJ, Sloot PMA, Brissos S, et al., editors. Computational science – iccs 2020. Cham: Springer International Publishing; 2020. pp. 524–537.

4. Pitler E, Bergsma S, Lin D, Church K. Using web-scale n-grams to improve base np parsing performance. 2010;

5. Feynman RP, Vernon Jr. FL. The theory of a general quantum system interacting with a linear dissipative system. Annals of Physics. 1963;24: 118–173. doi:10.1016/0003-4916(63)90068-X

6. Dirac PAM. The lorentz transformation and absolute time. Physica. 1953;19: 888–896. doi:10.1016/S0031-8914(53)80099-6