Capstone Public Project Summary

Exploration of the European Union Open Data Portal

Patrik Szigeti, MS in Business Analytics

Introduction of the problem

The primary objective of this capstone was to collect and explore the public data sources present on the European Union Open Data Portal to curate a list of data sources potentially useful for my client based on machine learning and natural language processing techniques. Data can be the greatest asset in the right hands, and finding the right datasets methodically and periodically can ensure a competitive advantage by keeping on top of current trends and staying in a position that allows for quick reactions, or possibly prediction of market changes.

The European Union Open Data Portal is a collection of more than 15 thousand publicly available datasets published by EU institutions and bodies. Though the Portal does have keywords and certain categorizations, there are no strict guidelines uploaders must adhere to, and the search functionality is limited – this combined with the large number of possibilities makes it hard for a human to find valuable data.

Together with my client, we have identified the real estate related areas of expertise they would like to focus on, and our expectation is that the Portal contains useful datasets for different countries and cities of interest that they would be able to leverage. The task at hand is to figure out the likelihood with which a dataset from the Portal will fall into any of the buckets we determined.

Questions to be answered

- First and foremost, I needed to determine whether the Portal is a good, valuable source to help achieve our goals? Is it worth spending time and resources on it?
- Do we need to refine the training data for the model by excluding certain keywords or by including texts from e.g. research papers in addition to Wikipedia articles?
- Are the data sources that were identified by the model on the right level of granularity?
- Are there any periodically uploaded datasets that the client can learn from on an ongoing basis?

The problem translated to the language of data

In order to answer these questions and find valuable datasets by applying NLP techniques:

1. I built out a training set which allows me to teach my model what keywords make up certain categories from our perspective.
2. I obtained metadata from the Portal so that I can use that as my holdout set.
3. I built out an incremental pipeline in Python so that the first two steps are easily reproducible.
4. I created a model in R that assigns probabilities for the three categories for each dataset by three different methods, then takes the average of these methods to come up with a final predicted number for each of them.
5. I created a Shiny dashboard in R to visualize these predictions and to allow users to further explore the datasets they deemed useful.

Training set – Wikipedia

The training dataset is created by scraping the contents of Wikipedia articles closely related to three areas we decided to focus on. These all have their own categories on Wikipedia with subcategories and directly tagged articles – for two of them, we included articles of the subcategories as well. I ended up scraping the contents of 1,516 articles altogether, which resulted in an imbalanced set that I had to account for while modeling to ensure I would not categorize everything to the same bucket because of its share from the population.

Holdout set – EU Open Data Portal

For my holdout set, I used the open API of the EU Open Data Portal. The calls to the API returned a rather noisy JSON file, so I had to spend time exploring the website and figuring out what attributes am I interested in. I pulled the name of the dataset for identification purposes, its description to feed to my model and its URL to ensure connection with the Portal for the end-users. All this is done incrementally, any newly uploaded datasets are appended to the golden source of metadata to avoid redundancy.

Modeling methodology

In order to make the two datasets compatible, I performed the same pre-processing techniques on both. I used tokenization to break the text into tokens, then filtered on those that occurred the most within the corpus. I removed stopwords such as *“and”*, *“are”*, *“or”* etc. as

these have no added value. Then I leveraged tf-idf (term frequency – inverse document frequency), which assigns higher weights to less common words throughout the document, therefore assigning higher importance to them.

I trained the model on three different sets of the data extracted from Wikipedia:

1. Only the summary excerpts – the first paragraph of each article.
2. The full article without the summary.
3. And the first two items combined.

This is a multiclass classification problem, and the output of each model run is a probability matrix for every dataset – what is the likelihood that an item falls into any of the three pre-determined categories. The matrices each contain three values between 0 and 1, and add up to 1 combined. This means that even though some datasets might have nothing to do with any of the target categories, they will still be assigned a probability based on the training data. Given the three approaches, I ended up with three probability matrices for each dataset that I flattened by calculating the grand mean, a commonly used measure for averaging probabilities.

I used a random forest model that achieved a 92-93% AUC on the test set after parameter-tuning, which means the possibility that it would position a randomly selected article into the right bucket was rather high. The final result set contains the names and descriptions of datasets, the probabilities for all three categories and links to the Portal.

Visualization of the results

It can be challenging to harness useful information from these Excel files, so I decided to create a Shiny dashboard to visualize the output of the model. The dashboard contains filters to specifically look for geospatial data or for surveys, as well as three different tabs:

- "Probabilities" to visualize the outcome from the model in a table format – this shows the top N results based on the user's selection for the chosen category, arranged in a descending order by their probabilities. Despite the high probabilities, we cannot be sure that something will automatically be useful for my client, so there is an additional functionality to save selected datasets.
- "Selected Datasets" shows a shortened list containing each dataset that was selected on the previous tab.

- The “Drilldown” tab serves as a sort of sneak peek into the datasets that were selected. Unfortunately, it does not work in each case due to the loose guidelines for uploading files to the Portal.

Recommendations

- It might be worth enriching the data obtained from the Portal with keywords, domains and concepts that are sparsely populated, but still could prove useful.
- While Wikipedia is a good starting point for the training data, to enhance the accuracy and the quality of the model, it could be a good idea to include research papers and publications from any of the target areas.
- The model itself can always be perfected, it could be interesting to play with other kinds of models, or maybe even to create an ensemble model by leveraging the best attributes from each possibility.

Summary

I have set out to create a list of potentially useful data sources, and to first and foremost determine whether the European Union Open Data Portal is a valuable source of data going forward. I believe I was able to prove that it is, and we managed to go from more than 15 thousand poorly labeled datasets to a handful, carefully selected items, that can potentially be leveraged for supporting my client’s core analysis. There are datasets on the right level of granularity, and some that are uploaded on a quarterly or yearly basis.

I believe this capstone showcases how to leverage NLP techniques and modeling to make sense of a rather noisy dataset, and I am hopeful that my client will be able to turn some of the datasets we discovered to their advantage.