Artificial Neural Networks Applied to Named Entity Recognition of Structured Data Sets

Master's Thesis, Submitted on

Miriam Herman
Yeshiva University - Department of Mathematical Sciences
215 Lexington Ave
New York, NY
miriam.herman@mail.yu.edu

ABSTRACT

This paper demonstrates the utilization of Artificial Neural Networks (ANNs) to classify the contents of column data in structured data sets. Using pre-trained word embeddings from GloVE and Word2Vec, we demonstrate ____ improvement over the Stanford NLP and ____ baseline for the task of classifying names, organizations, and addresses. We also introduce a classifier for products. Our best model has ____ percent accuracy on unseen sets.

CCS Concepts

•**Computer systems organization** → **Embedded systems;** *Redundancy;* Robotics; •**Networks** → Network reliability;

Keywords

ACM proceedings; LaTeX; text tagging

1. INTRODUCTION

According to a survey conducted by CrowdFlower, a platform for data scientists, "data preparation accounts for about 51% of the work of data scientists" and "60 % of data scientists view data preparation [collecting, labeling, cleaning, and organizing data] as the least enjoyable part of their work." [1] The report also stated that 49 % of a data scientists work involves structured data. It would therefore be extremely valuable to provide a rigorous method for tagging the contents of structured data sets to minimize the amount of time data scientists must spend on the tasks they enjoy the least.

We compare the results of our state of the art ANN models to baselines from Stanford NLP, a simple count based bag-of-words, and ____ to find a ____ percent improvement. Our source code and training sets can be found at ____.

WOODSTOCK '97 El Paso, Texas USA

DOI: 10.475/123_4

2. THE *BODY* OF THE PAPER

Typically, the body of a paper is organized into a hierarchical structure, with numbered or unnumbered headings for sections, subsections, sub-subsections, and even smaller sections. The command `\section` that precedes this paragraph is part of such a hierarchy.[1] LaTeX handles the numbering and placement of these headings for you, when you use the appropriate heading commands around the titles of the headings. If you want a sub-subsection or smaller part to be unnumbered in your output, simply append an asterisk to the command name. Examples of both numbered and unnumbered headings will appear throughout the balance of this sample document.

Because the entire article is contained in the **document** environment, you can indicate the start of a new paragraph with a blank line in your input file; that is why this sentence forms a separate paragraph.

2.1 Data Collection and Cleansing

Data for training and testing was acquired from a variety of sources, with duplicates removed. The appendix contains a complete list of data sources used, with notes regarding file name, column name and any additional steps taken on the data.

Once all the data was collected, the data was divided by type and compiled for each type into a single file. All duplicates were removed. The final count includes 723,553 unique addresses, 1,061,544 unique companies, 249,227 unique peoples names and 559,591 unique products. The data sets were then shuffled, and 20% of each set was removed for testing purposes.

We also generated a set of 250,000 names using a datafactory that utilizes US census information to create name data determined by name frequency. We perform tests and training on names data with and without the generated names.

2.2 Baselines

In order to assess the necessity of a Neural Network trained classifiers for structured data, three baseline measures were initially tested; Stanford NLP NER [2], openNLP, and python's NLTK package.

We tested the models on address, company, and names

[1]This is the second footnote. It starts a series of three footnotes that add nothing informational, but just give an idea of how footnotes work and look. It is a wordy one, just so you see how a longish one plays out.

data, using the following metric. A tag was labeled correct if all words in a cell had the same label, and that label was correct, otherwise it was labeled incorrect. If the model allowed, we also show the metric for all words in a cell having at least once the correct label, and the rest "other."

The results of each model are below.

2.2.1 Stanford NLP NER

We used the Stanford NLP NER tool to classify addresses, organizations and people, but as right from the box it is only capable of classifying 7 specific types: Location, Person, Organization, Money, Percent, Date, Time, we did not use it to classify product data.

Using the above metric, we found the Stanford NLP could not classify addresses. It wasn't able to classify addresses because it classified items crucial to identifying addresses, such as street numbers and words like "Rd" and "Ave" as "Other". A typical example of classifier output is: 603/O HINMAN/PERSON RD/O where "603" and "RD" were labeled as "Other" and "FINMAN" labeled as "PERSON". For the 144,709 addresses in the test set, the model had 0% accuracy, labeling only 2 address correctly. If we consider the metric of accuracy to be "correct label" and "other" the accuracy on this set increases to 9.28% or 13,431 addresses labeled correctly.

We also found the Stanford NLP NER classifier struggled to classify organizations for a very similar reason the model failed to classify addresses: words crucial to an organization, such as "Limited", are classified as "Other". A typical example of the output for an organization cell is: ALASTAIR/PERSON WRAY/PERSON LIMITED/O. Since many companies are named after their founder, there are often confused for names. For the 212,307 companies in the test set, the model had 0.018 % accuracy, labeling only 38 companies correctly.If we consider the metric of accuracy to be "correct label" and "other" the accuracy on this set increases to 8.95% or 19,009 companies labeled correctly.

Since the Stanford NLP NER classifier does not distinguish a single letter as a name, and because a lot of our name data consists of first and middle names listed with just an initial, the Stanford NLP NER struggled to classify people as well. A typical example of the output for a name is:I/O SONI/PERSON, where the initial is labeled as "Other". Of the 49,844 names in the test set, the model had 3.66 % accuracy, labeling 1825 examples correctly. If we consider the metric of accuracy to be "correct label" and "other" the accuracy on this set increases to 52.73% or 26,287 names labeled correctly.

When the Stanford NLP NER classifier was tested on the collected names data together with the datafactory generated names data, it performed significantly better, labeling 39.69% or 35,634 out of 89,789 names correctly. (Missing 2486 generated names.) If we consider the metric of accuracy to be "correct label" and "other" the accuracy on this set increases to 71.89% or 64,548 names labeled correctly.

Out of 406,860 examples, only 1865 or 0.5% of examples were labeled correctly.

2.2.2 openNLP

We used the OpenNLP NER tool to classify addresses, organizations and people, but like StanfordNLP, it is only capable of classifying; Location, Person, Organization, Money, Percent, Date, Time right from the box. We did not attempt to train it to classify product data.

OpenNLP performed much better than Stanford NLP when classifying addresses because it seems to recognize certain words common to addresses, like "Ave" and "St". However, it does not recognize other crucial words, like "Rd"s and misclassified most "Rd"s as "Organizations". Out of 144,709 addresses in the test set, openNLP classified 19.22% or 27,818 examples correctly.

This model also performed much better than Stanford NLP when classifying companies, recognizing common company abbreviations like "LTD" and "CO" as part of an organization. However, it failed to recognize common company words like "Limited" and "Agency". Out of 212,307 companies in the test set, openNLP classified 36.39% or 77,265 examples correctly.

When classifying names openNLP performed significantly better than Stanford NLP because it classified full terms, and didn't classify names with initials if the last name was recognized strongly as a name. A name like "A G QUITO," with an uncommon last name, would have been misclassified as an organization and some names that were not formatted well were left unclassified, such as "B HARRIS". Out of 49,844 names in the test set, openNLP classified 63.16% or 31,483 examples correctly.

When the openNLP NER classifier was tested on the collected names data together with the datafactory generated names data, it performed slightly better, labeling 70.58% or 63,374 out of 89,789 names correctly.

Out of 406,860 examples openNLP classified a total of 136,576 or 33.6% examples correctly.

2.2.3 NLTK Classifer

We attempted to classify addresses, organizations and people using the Natural Language Toolkit (NLTK) package in Python. The NLTK package can classify; Organization, Person, Location (for example Mount Everest), Date, Time, Money, Percent, Facility, GPE (Geo-political entities, like city, state/province, country). We used the toolkit straight from the box, using it to classify by Organization, Person, and GPE and did not attempt to train it to classify product data.

NLTK performed much better than Stanford NLP when classifying addresses but worse than openNLP. It seems to recognize certain words common to addresses, like "East" and "Indian" but not the crucial words like "st", "rd", and "ave". The NLTK classifier misclassified addresses as organizations, people, and other, but very rarely confused them with several mixed labels. Out of 144709 address in the test set, NLTK classified 2.61% or 3,771 examples correctly.

This model performed much better than Stanford NLP and openNLP when classifying companies, recognizing common company abbreviations like "LTD" and "CO" as part of an organization. It is hard to understand why it classified some companies incorrectly. Out of 212,307 companies in the test set, NLTK classified 82.73% or 175,631 examples correctly.

When classifying names NLTK performed worse than Stanford NLP and openNLP, and it is unclear why. On occasion NLTK would correctly identify a name with initials as a person, and in other cases it would classify them as organizations, GPEs, or other. There does seem to be a correct trend of classifying names as "Person" if there is a first and middle initial, but this is not consistent. Out of 49,844 names

Table 1: Stanford NLP Confusion Matrix

	Address	Company	Name	Product	Other	Mixed
Other						
Address	2	0	41	NA	36,101	108565
Company	3	38	324	NA	121,613	90,329
Name	0	2	1825	NA	6955	41,062
Product	NA	NA	NA	NA	NA	NA

Table 2: openNLP Confusion Matrix

	Address	Company	Name	Product	Unclassified	Mixed
Address	27,818	37,145	5759	NA	10,971	63,016
Company	268	77,275	578	NA	442	133,744
Name	131	611	31,483	NA	3060	14,559
Product	NA	NA	NA	NA	NA	NA

in the test set, openNLP classified 3.46% or 1727 examples correctly.

When the NLTK classifier was tested on the collected names data together with the datafactory generated names data, it performed better, labeling 20.43% or 18,346 out of 89,789 names correctly.

Out of 406,860 examples openNLP classified a total of 181,129 or 44.52% examples correctly. (This increase in total accuracy is due to NLTK's proficiency at classifying companies.)

2.3 Neural Network Training

In my previous work a thorough analysis of embeddings, including Facebook fastText, word2vec, continuous bag of words dependency based embeddings, character embeddings, and more, was done. We learned that the best pre-trained embeddings for this task are the 100 dimensions GloVe word vectors. Our models were trained using this embeddings.

We set out intending to perform a grid search for the parameters of our model. We began our tests with the simplest models, creating binary models for each data type. Each model had a single layer and 128 nodes, was optimized with adam and trained with 10 epochs. The four models achieved 97%-99% accuracy. As this was huge improvement from all the previous baselines, and because no model will have 100% accuracy, we did not see a need to fine tune the model further.

Our models were trained on 2,016,028 unique phrases containing 389,808 unique words. The max number of words in a phrase was decided to be 10, so our model was trained on 10 dimensional word vectors. If a phrase had fewer than 10 words, it's associated vector was padded with zeros. We trained two models, one with the datafactory generated names and one without. Both models were subsequently tested on our test set and on a generated business data set. The models generated probability predictions for the each type. The results are presented.

2.3.1 Test Set Results

We tested the model on the same set of data the baseline classifiers were tested on. We tested two models, one with generated names and one without, and for all the data types aside from names, the two models performed almost identically. For names, the model trained with the generated names performed 4% better, from 94% to 98%, and therefore going forward in this paper we will train models on the generated names. The address classifier hovered around 99%, the companies classifier around 98% and the product classifier at 96%.

2.3.2 Business Set Results

When tested on the business data set, the model performed well on names and companies, however it struggled to classify addresses and products. This result is partly due to the data in the spreadsheet, for example in the column"street" words like "st" and "rd" were omitted. However, the model also struggled with the "billing address" column that had entries like "Suite # 102239", only classifying it with 63% confidence as an address.

The model especially struggled classifying dates, labeling them as products with almost 100% confidence. Also, due to single letter initials in the names data, entries with single letters, like "pay cycle" were classified as a name with 98% likelihood.

We may improve our results by using an LSTM (Long Short Term Memory) Model, for then our algorithm will be trained on not just a single word, but a sequence of them.

2.4 LIME - Explaining the Predictions

Machine learning algorithms are black box solutions, where input is fed to a model and a classification returned without explanation as to how the decision was made. LIME, LOCAL INTERPRETABLE MODEL-AGNOSTIC EXPLANATIONS [**?**], attempts to solve this problem by perturbing data, running the model on it, and seeing the classification probability result.

We provided LIME with 100 examples each of addresses, companies, products, and names, and it returned the 6 most important terms used in the classification process. The results are listed below. As was expected, Ave, Rd, and St are the most important words when classifying an address. That ACTUARIAL and CJH are the most important for companies is surprising, and this could be a reflection of the 100 specific examples chosen for testing. The results of the Product test are very telling, for very surprising terms like "500" and "2B" carry a lot of weight.

2.5 Tables

Because tables cannot be split across pages, the best placement for them is typically the top of the page nearest their

Table 3: NLTK Confusion Matrix

	Address	Company	Name	Product	Other	Mixed
Address	3771	30,096	45,135	NA	64,954	758
Company	733	175,631	1098	NA	19,958	14,887
Name	117	7218	1727	NA	39,498	1284
Product	NA	NA	NA	NA	NA	NA

Table 4: Generated Names NN Confusion Matrix

	Address	Company	Name	Product
Address	.997448	.0017542	.0003253	.003252
Company	.00128143	.987642	.002799	.007761
Name	.0010614	.008826	.989925	.01163
Product	.00407549	.0178114	.006845	.965432

Table 5: Collected Names NN Confusion Matrix

	Address	Company	Name	Product
Address	.996454	.001618	.000358	.00376
Company	.000656	.987531	.002567	.007580
Name	.006821	.0619743	.949441	.0271665
Product	.00291918	.0184617	.0068049	.965063

initial cite. To ensure this proper "floating" placement of tables, use the environment **table** to enclose the table's contents and the table caption. The contents of the table itself must go in the **tabular** environment, to be aligned properly in rows and columns, with the desired horizontal and vertical rules. Again, detailed instructions on **tabular** material is found in the *LaTeX User's Guide*.

Immediately following this sentence is the point at which Table 1 is included in the input file; compare the placement of the table here with the table in the printed dvi output of this document.

To set a wider table, which takes up the whole width of the page's live area, use the environment **table*** to enclose the table's contents and the table caption. As with a single-column table, this wide table will "float" to a location deemed more desirable. Immediately following this sentence is the point at which Table 2 is included in the input file; again, it is instructive to compare the placement of the table here with the table in the printed dvi output of this document.

2.6 Figures

Like tables, figures cannot be split across pages; the best placement for them is typically the top or the bottom of the page nearest their initial cite. To ensure this proper "floating" placement of figures, use the environment **figure** to enclose the figure and its caption.

This sample document contains examples of **.eps** files to be displayable with LaTeX. If you work with pdfLaTeX, use files in the **.pdf** format. Note that most modern TeX system will convert **.eps** to **.pdf** for you on the fly. More details on each of these is found in the *Author's Guide*.

As was the case with tables, you may want a figure that spans two columns. To do this, and still to ensure proper "floating" placement of tables, use the environment **figure*** to enclose the figure and its caption. and don't forget to end the environment with figure*, not figure!

Figure 1: A sample black and white graphic.

Figure 2: A sample black and white graphic that has been resized with the `includegraphics` command.

2.7 Theorem-like Constructs

Other common constructs that may occur in your article are the forms for logical constructs like theorems, axioms, corollaries and proofs. There are two forms, one produced by the command `\newtheorem` and the other by the command `\newdef`; perhaps the clearest and easiest way to distinguish them is to compare the two in the output of this sample document:

This uses the **theorem** environment, created by the `\newtheorem` command:

THEOREM 1. *Let f be continuous on $[a, b]$. If G is an antiderivative for f on $[a, b]$, then*

$$\int_a^b f(t)dt = G(b) - G(a).$$

The other uses the **definition** environment, created by the `\newdef` command:

Definition 1. If z is irrational, then by e^z we mean the unique number which has logarithm z:

$$\log e^z = z$$

Two lists of constructs that use one of these forms is given in the *Author's Guidelines*.

There is one other similar construct environment, which is already set up for you; i.e. you must *not* use a `\newdef` command to create it: the **proof** environment. Here is a example of its use:

PROOF. Suppose on the contrary there exists a real number L such that

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = L.$$

Then

$$l = \lim_{x\to c} f(x) = \lim_{x\to c} \left[gx \cdot \frac{f(x)}{g(x)} \right] = \lim_{x\to c} g(x) \cdot \lim_{x\to c} \frac{f(x)}{g(x)} = 0 \cdot L = 0,$$

Table 6: Business Data Results

Col Name	Example	Classification	Confidence
Account Owner	Toni Gomez	Person	0.998
Billing Contact	Allen Hardin	.999	.000358
Billing Email	jgoff@ma1l2u.org	company	.826
City	Helena	Person	.429
Conversion Date	Thu Apr 03 05:20:32 EDT 2014	Product	.999
Country	Slovakia	Product	.430
Custom Metrics	code,text,room	Product	.584
Org Name	Morgan Studios	Company	.864
Parent Name	'Ringgold Cafe'	Company	.861
Pay Cycle	'Q','T'	Person	.982
Pay Method	'Invoice'	Product	.472
po_num	PO6793946273	Company	.575
State	'in', 'so'	Product	.664
Street	'Mcconnel'	Person	.501
Terms	'Standard'	Product	.661
Valid From	Thu Jul 30 05:20:32 EDT 2013	Product	.999
Valid To	Sat Mar 15 05:20:32 EDT 2014	Product	.999

Table 7: LIME - Most Important Words

Data Type	6 Important Words
Address	Saddle, AVE, Rd, Suffield, MADISON, St
Company	LIMITED, PROPERTY, ACTUARIAL, CJH, PLAYING, WORLD
Name	SINGH, Frank, GALLIMORE, Harding, Bush, Remirez
Product	500, Coffee, Non-Stick, 2B, Wine, rigolo

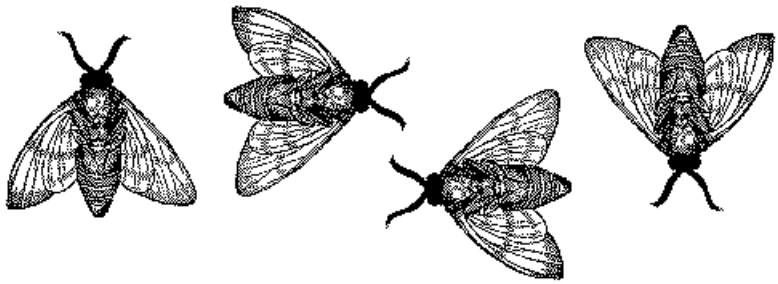

Figure 3: A sample black and white graphic that needs to span two columns of text.

Table 8: Frequency of Special Characters

Non-English or Math	Frequency	Comments
Ø	1 in 1,000	For Swedish names
π	1 in 5	Common in math
\$	4 in 5	Used in business
Ψ_1^2	1 in 40,000	Unexplained usage

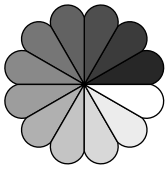

Figure 4: A sample black and white graphic that has been resized with the `includegraphics` command.

which contradicts our assumption that $l \neq 0$. □

Complete rules about using these environments and using the two different creation commands are in the *Author's Guide*; please consult it for more detailed instructions. If you need to use another construct, not listed therein, which you want to have the same formatting as the Theorem or the Definition[3] shown above, use the `\newtheorem` or the `\newdef` command, respectively, to create it.

A *Caveat* for the TEX Expert

Because you have just been given permission to use the `\newdef` command to create a new form, you might think you can use TEX's `\def` to create a new command: *Please refrain from doing this!* Remember that your LATEX source code is primarily intended to create camera-ready copy, but may be converted to other forms – e.g. HTML. If you inadvertently omit some or all of the `\def`s recompilation will be, to say the least, problematic.

3. CONCLUSIONS

This paragraph will end the body of this sample document. Remember that you might still have Acknowledgments or Appendices; brief samples of these follow. There is still the Bibliography to deal with; and we will make a disclaimer about that here: with the exception of the reference to the LATEX book, the citations in this paper are to articles which have nothing to do with the present subject and are used as examples only.

4. ACKNOWLEDGMENTS

This section is optional; it is a location for you to acknowledge grants, funding, editing assistance and what have you. In the present case, for example, the authors would like to thank Gerald Murray of ACM for his help in codifying this *Author's Guide* and the **.cls** and **.tex** files that it describes.

5. REFERENCES

[1] 2017 data scientist report. crowdflower.com, 2017.

[2] T. G. Jenny Rose Finkel and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. pages 363–370. ACM, 2005.

[3] S. Salas and E. Hille. *Calculus: One and Several Variable*. John Wiley and Sons, New York, 1978.

APPENDIX

A. HEADINGS IN APPENDICES

The rules about hierarchical headings discussed above for the body of the article are different in the appendices. In the **appendix** environment, the command **section** is used to indicate the start of each Appendix, with alphabetic order designation (i.e. the first is A, the second B, etc.) and a title (if you include one). So, if you need hierarchical structure *within* an Appendix, start with **subsection** as the highest level. Here is an outline of the body of this document in Appendix-appropriate form:

A.1 Introduction

A.2 The Body of the Paper

A.2.1 Type Changes and Special Characters

A.2.2 Math Equations

Inline (In-text) Equations.

Display Equations.

A.2.3 Citations

A.2.4 Tables

A.2.5 Figures

A.2.6 Theorem-like Constructs

A Caveat for the TEX Expert

A.3 Conclusions

A.4 Acknowledgments

A.5 Additional Authors

This section is inserted by LATEX; you do not insert it. You just add the names and information in the `\additionalauthors` command at the start of the document.

A.6 References

Generated by bibtex from your .bib file. Run latex, then bibtex, then latex twice (to resolve references) to create the .bbl file. Insert that .bbl file into the .tex source file and comment out the command `\thebibliography`.

B. MORE HELP FOR THE HARDY

The sig-alternate.cls file itself is chock-full of succinct and helpful comments. If you consider yourself a moderately experienced to expert user of LATEX, you may find reading it useful but please remember not to change it.

Table 9: Some Typical Commands

Command	A Number	Comments
`\alignauthor`	100	Author alignment
`\numberofauthors`	200	Author enumeration
`\table`	300	For tables
`\table*`	400	For wider tables

Table 10: Data Sources

Source	Data Type	# records	Notes
openaddresses.io	Addresses	707094	Concatenated number and street from US Northeast
sec.gov /rules/other/4-460list.htm	Companies	951	Removed headers
qualityfrozenfoods.com	Products	5615	Downloads
crownprod.com/	Products	5615	Downloads
ikea.com	Products	2765	copied all products from site catalog
product-open-data.com/	Products	551953	GTIN table GTIN_NM column
product-open-data.com/	Companies	4151	brand table brand_NM column
wordlab.com	Companies	4924	company-names-list Removed top 66 and bottom 33 rows
wikipedia	Companies	688	List_of_common _carrier_freight _railroads_in _the_United_States
wikipedi	Companies	1648	List_of_companies _of_the_United_States
wikipedia	Companies	103	List_of_department _stores_of_the _United_States
wikipedia	Companies	112	List_of_independent _bookstores _in_the_United_States all listed with"in city"
wikipedia	Companies	404	List_of_supermarket _chains _in_the_United_States split on '(' and '-'
wikipedia	Companies	66	List_of_United_States _clock_companies
wikipedia	Companies	259	List_of_United_States _insurance_companies
wikipedi	Companies	502	List_of_United_States _water_companies
census.gov	People	5494	1990_census_namefiles Names generated with most common first names concatenated with most common last names
data.gov	Companies	1139	Active_Benefit _Companies Business Name
data.gov	People	237260	Civil_List Name
data.gov	Companies	4362	Consumer_Complaints Company
data.gov	Products	18	Consumer_Complaints Product
data.gov	Addresses	2112	FOIL_Report - Trade_Waste_All _Approved_or_Denied Mailing Office

Table 11: Data Sources part 2

Source	Data Type	# records	Notes
data.gov	Companies	2151	FOIL_Report - Trade_Waste_All _Approved_or_Denied - Trade Name
data.gov	Addresses	369	IDOL_2013 _Registered_Owner _Rideṡcsv Address
data.gov	Companies	1100	IDOL_2013 _Registered_Owner _Rideṡcsv Ownername and Manufacturer
data.gov	People	367	IDOL_2013 _Registered_Owner _Rideṡcsv ContactName
data.gov	Products	1731	IDOL_2013 _Registered_Owner _Rideṡcsv Ridename
data.gov	Companies	1504	Licensed _Insurance _Companies Company Name
data.gov	Addresses	488	Lobbying _Reporting _System _Maryland _Registered _Employers _List _ Address
data.gov	Companies	444	Lobbying _Reporting _System _Maryland _Registered _Employers _List _ Firm Name
data.gov	People	652	Lobbying _Reporting _System _Maryland _Registered _Employers _List _ Concatenate(First Name, Middle Name, Last Name)
data.gov	Companies	184	Lobbyist _Activity _Contacts Lobbyist Firm
data.gov	Addresses	270	Lobbyist _Activity _Contacts_ Lobbyist
data.gov	Addresses	5031	M_WBE_LBE _and_EBE _Certified_Business_List-Address1
data.gov	People	5327	M_WBE_LBE _and_EBE _Certified_Business_List-Contact_Name
data.gov	Companies	5410	M_WBE_LBE _and_EBE _Certified_Business_List-Vendor_Formal _Name
data.gov	Addresses	200	Neighborhood _and_Rural _Preservation _Companies _Directory Street Address

Table 12: Data Sources part 3

Source	Data Type	# records	Notes
data.gov	Companies	205	Neighborhood _and_Rural _Preservation _Companies _Directory Organization Name
data.gov	Products	2306	nsn _extract _4518-Common_Name
data.gov	Addresses	7320	Oregon _Consumer _Complaints Address 1
data.gov	Addresses	1038	Oregon _Consumer _Complaints Address 2
data.gov	Companies	9403	Oregon _Consumer _Complaints
data.gov	Companies	38	OWEB _Small _Grant _Teams Team
data.gov	People	38	OWEB _Small _Grant _Teams Team Contact
data.gov	Companies	1798	Prequalified _Firms - Prequalified Vendor Name
data.gov	Companies	266	SCA _Disqualified _Firms Vendor Name
data.gov	Companies	47	Science _Festival _Company _Sponsors Company Sponsor
data.gov	Companies	58	Top _50 _Employers _ _Hawaii _County - Name
data.gov	People	54	Top _50 _Employers _ _Hawaii _County Concatenate (Contact First Name, Contact Last Name)
data.gov	Addresses	135	Top _Manufacturing _Companies _in _SSMA _Region Primary Address
data.gov	Companies	202	Top _Manufacturing _Companies _in _SSMA _Region Company Name and Ultimate Parent
data.gov	People	233	Top _Manufacturing _Companies _in _SSMA _Region First Name and Last Name
data.gov	Addresses	71	Trade _Waste _Broker _Registrants Address
data.gov	Companies	71	Trade _Waste _Broker _Registrants Account Name
DBpedia	Companies	82838	Column B
DBpedia	People	200946	
data.gov.uk	Companies	1,045,333	BasicCompanyData