

Duplicate recognition for restaurant dataset*

1st Alexander Christoph

OTH Regensburg

Regensburg, Germany

alexander.christoph@st.oth-regensburg.de

Abstract—This document describes the analysis and removal of duplicates from the restaurants dataset. The aim is to remove as many duplicates as possible from the dataset and store the data without duplicates in a cloud hosted mongodb instance. A problem when finding duplicates of restaurants (or nearly any other dataset) is the format and the different writing of the entries in the data. This problem was already researched by several IEEE members (quelle). Within my research I used different approaches to preprocess and detect duplicates. The accuracy of my results is measured with the metrics accuracy, precision and recall. After removing the duplicates the cleared dataset is stored into a mongodb cluster so that it can be accessed any time.

Index Terms—duplicate detection, restaurant dataset

I. INTRODUCTION

In times where big data gets more and more attraction from the industry it is very important to learn how to deal with it. Especially when it comes to the structure and format of the data. When looking at big data, it is most of the time a problem that there are duplicates and unclean entries in a dataset. This gives inaccurate results when analyzing or working with that data. This happens because most of the time there isn't that much preprocessing happening and there aren't even checks for a standard data format. To learn how to deal with duplicate data, the restaurants dataset is used. This isn't actually big data, but to understand the importance of the preprocessing task it is pretty good because it's considered a well researched dataset to play with and compare to the gold standard.

In my research I've looked into different approaches to detect duplicates and remove them from the dataset. The first approach was to just remove all duplicates that are in the data, this wasn't successful because most of the duplicated entries had different writing or completely different values in some of the fields. So I started to analyze the data and look for potential duplicates and how to prepare them so that the program can match them. The first step was to remove all special characters and some other unnecessary contents in the different columns. After that I investigated which columns are the most useful when it comes to duplicate detection and researched them more accurate to prepare them correctly.

After each step of my preprocessing pipeline, I calculated different values and metrics for my potential duplicates. These were: the count for true positives, false positives, true negatives and false negatives as well as the metrics accuracy, recall and precision. To calculate these, I used the gold standard duplicate dataset which was evaluated by hand. With the help

of these metrics I was able to decide how good or bad my results were. As a conclusion the best values I archived were:

- All entries in original dataset: 864
- Detected duplicates (all): 100
- Real duplicates (from gold standard): 112
- True positives: 100
- True negatives: 752
- False positives: 0
- False negatives: 12
- Accuracy 0.97
- Precision: 1.0
- Recall: 0.89

After the methods were applied and duplicates were removed it was necessary to store the new dataset somewhere. For this I have chosen mongodb because of it's great compatibility with many programming languages and the low expenses when you want to store data in it. Mongodb could also be used for many preprocessing tasks because of the great aggregation framework that it offers. In this research I rather focused on the preprocessing with pandas, a python data science library.

II. WHY DUPLICATE DETECTION IS IMPORTANT

Duplicate detection is an important task in data science for several reasons. Often when a data scientist gets data to research, it's a problem that it isn't preprocessed and contains wrong or duplicate entries. Detecting and removing these is a hard but important task. When working with unclean data, mistakes can happen. For example let's look at the restaurant dataset which is described in *section III The restaurants dataset*. This data could be used to recommend restaurants to an user of an application. When the recommendation contains duplicates with even different writing for the same restaurant, the user could be confused and stop using our application. Duplicate detection is also important when solving machine learning tasks because most machine learning algorithms can't handle duplicates and because of this give higher priority to duplicated records which would result in a less accurate machine learning model.

Because of these problems, duplicate detection is such a big topic in the computer science space.

III. THE RESTAURANTS DATASET

The restaurants dataset which was researched in this paper is a .tsv dataset that describe the location and type of different

restaurants. It contains 864 rows of data with six columns. The columns of the dataset are:

- id: The unique id of each row
- name: The name of the restaurant
- address: The address where the restaurant is located
- city: The city of the restaurant
- phone: The phone number of the restaurant
- type: The kind of the restaurant (i.e. french or american)

In the data there are 122 duplicated restaurants. These duplicates were picked by hand from some researchers to define a gold standard which would be the best possible result after removing all duplicates. These duplicates have different deviations from each other. For example some have a different order of the words in their name field like "the palm" and "palm the". Others have different separators for the phone number like "310/659-9639" and "310-659-9639". Sometimes the city field of the duplicates is a district from a bigger city and sometimes it's the city name itself like "los angeles" and "hollywood". There are even more different deviations in the dataset whose solution to detect them will be discussed later in *section V Methods used in the preprocessing task* section.

In my research I focused mostly on the columns name, city, address and phone because they have the most useful information when it comes to duplicate detection. The id column was left out because it's only a unique identifier which wouldn't bring a benefit for duplicate recognition. The type column was left out because there were too much restaurants with the same type which would result in an unclear target data record.

Besides of the plain restaurants dataset there is also a dataset given which contains all the duplicates of the data by id. In this duplicate set, there are only two columns, "id1" and "id2" which define the original id and the duplicate id. A dataset without all these duplicates is considered as gold standard. The reached results will be measured to this gold standard.

IV. METRICS USED TO MEASURE RESULTS

To measure the accuracy of results when it comes to duplicate detection there are different metrics that are commonly used. For all these it's important to pre calculate four values with the help of the gold standard data and the archived results:

- True positives (TP): The correct classified true entries
- True negatives (TN): The correct classified false entries
- False positives (FP): Incorrect classified true entries
- False negatives (FN): Incorrect classified false entries

In a gold standard dataset there are no false positives and false negatives.

A first metric that is important is the well known accuracy.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Only calculating the accuracy isn't enough in this task because the it only gives reliable results when a dataset is balanced. In this case it means that there are as many duplicated entries as non duplicates. As an addition for the accuracy I use two other metrics, precision and recall.

Precision measures the exactness for the minority class by only considering the positive classified entries of the result set. Because of this it is a good measurement for unbalanced data.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Recall instead gives the accuracy for the fraction of relevant data.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

V. METHODS USED IN THE PREPROCESSING TASK

The research was written in python, a programming language, which is widely used for data science tasks. For my research I mostly used the Python pandas library [1], which has good data analysis features.

The goal of the methods used was to minimize the number of unique values in each column that represent the same "real" value so that an equality match for different column combinations(see ?? ??) can match them as duplicates.

A. Studying the dataset

The first step when it comes to data science tasks is to explore the structure and the different columns of a dataset.

This was done by viewing at the first lines of the dataset itself and looking at the values of the different entries to understand their meaning. After doing this, it's necessary to look into each column by itself. For this there was especially looked at the unique values of each column. Here it was found that for example the "type" column is very inconsistent with its values(i.e. sometimes the value is "pizza" and sometimes it's "italian" for the same kind of restaurant) and wouldn't fit as a column to find duplicates because there are too many restaurants with the same type after normalizing them. So the "type" column was dropped for this research.

The columns left ("name", "address", "city" and "phone")

The outcomes of researching the other columns name, phone and address are described in the subsections below

PLACEHOLDER!!!!

B. Step 1: Generic clearing of the columns

As a first approach it was necessary to do as many generic operations for all columns as possible.

Many noise in the data came from special characters. So I removed every special character except spaces from the four columns("name", "address", "city", "phone") that I looked into. Large and lower case letters didn't have to be considered because the whole data was in lower case letters. Otherwise it would have been necessary to convert all strings to lowercase.

Further examination of the columns has shown that some values contain cardinal directions where others don't. Sometimes the cardinal directions were even written differently (i.e. "n" and "north"). So the next step was to write a regex that detects every occurrence of a cardinal directions. The regex to detect these is:

'((|^)((south)|(east)|(west)|(north)|(ne)

This was then used to replace everything that matches with an empty string.

C. Clearing the city column

After the generic clearing, it was necessary to clear the specifics of each column by itself. It was discovered that some city names were written inconsistent. For example, sometimes "la" was written instead of "los angeles". So I mapped every double occurrence of a city name to a standard value. This map looked like:

"la", "west la" → "los angeles"
 "new york city" → "new york"

After applying this mapping, the "city" column can be considered as cleared.

VI. CLEARING THE ADDRESS COLUMN

As next step the address column needed to be cleared. I looked deeper into the address column and found out that some addresses have appendixes after the address that describes the location of the restaurant(i.e. "3125 piedmont rd. near peachtree rd."). This wasn't consistent either. So I decided to remove every appendix from the address column after the words "between", "off", "near", "at" and "in".

Then I discovered that there are different writings for numbers (i.e. "1", "1st" or "first"). So I removed every appendix of a number("st", "nd" and "rd") and mapped each number so that they are consistent. I have chosen to take the integer representation as a goal value.

Another inconsistency in the address column was the different writing of some words. So I mapped these different representations of the same words like:

"la" → "los angeles"
 "ave" → "avenue"
 "rd" → "road"
 "blv", "blvs" → "boulevard"
 "st" → "street"

This resulted in the new metrics:

So, one more duplicate was found by clearing the "address" column.

The next column to clear was the "name" column. at first I did a similar approach like I have done with the addresses, to remove every appendix after the words: "between", "off", "near", "at", "in" and "of". Then I discovered that sometimes the name of a restaurant also contains the city where it's located. This information is already present in the city column, so I decided to also remove these city names. I removed the occurred cities "los angeles", "las vegas", "new york city" and "new york". I also decided to remove some unnecessary words in the name field because they only brought noise in the data. These words were "the", "restaurant" and "and". This resulted in the new metrics:

After clearing the name, three more duplicates were found.

As a last step of my duplicate detection research I decided to sort the single words of the name column because often the

TABLE I
METRICS FOR THE REACHED RESULTS

Metric	step 1	step 2	step 3	step 4	step 5
Detected duplicates	1	1	1	1	5
True Positives	1	1	1	1	5
True Negatives	1	1	1	1	5
False Positives	1	1	1	1	5
False Negatives	1	1	1	1	5
Accuracy	1	1	1	1	5
Precision	1	1	1	1	5
Recall	1	1	1	1	5

restaurant names were written in a different word order. The sorting was a simple descending sort of all words in each row. This resulted in the final metrics:

In the final results there were 12 False Negatives left, which is pretty good. The most important is, that there were no False Positives, which would result in a bad precision and so would break the sense of duplicate detection. There are other algorithms to improve the archived results which will be discussed in *section IX Potential improvements and other researches*.

VII. RESULTS

VIII. STORING DATA INTO MONGODB

After doing my research and reaching a reliable result, it's important to store the resulted data somewhere. For this I have chosen to use a cloud hosted mongodb database because it's easy to use and fill databases from python. With mongodb it's possible to simply push python dictionaries to the database and it stores the data in a correct way. The cloud hosted approach was chosen to make it easy to access the data anytime.

Before storing the non duplicate dataset in mongodb, I had to change my resulted entries back to the original values because in the preprocessing task there were many changes to the data, which nearly made it unreadable for humans. For this I have chosen to select the first occurring entry for a duplicate from the original dataset.

IX. POTENTIAL IMPROVEMENTS AND OTHER RESEARCHES

Another improvement to this map would have been a mapping from districts to city names like "hollywood" → "los angeles", because the entries sometimes had the district and sometimes had the city name in its column.

X. CONCLUSION

REFERENCES

- [1] Python pandas library documentation, Accessed on: January 3, 2020 [online]. Available: <https://pandas.pydata.org/pandas-docs/stable/>
- [2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
- [3] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
- [4] K. Elissa, "Title of paper if known," unpublished.
- [5] R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

- [6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetism Japan, p. 301, 1982].
- [7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

XI. INTRODUCTION

This document is a model and instructions for \LaTeX . Please observe the conference page limits.

XII. EASE OF USE

A. Maintaining the Integrity of the Specifications

The IEEEtran class file is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

XIII. PREPARE YOUR PAPER BEFORE STYLING

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organizational editing before formatting. Please note sections XIII-A–XIII-E below for more information on proofreading, spelling and grammar.

Keep your text and graphic files separate until after the text has been formatted and styled. Do not number text heads— \LaTeX will do that for you.

A. Abbreviations and Acronyms

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, ac, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

B. Units

- Use either SI (MKS) or CGS as primary units. (SI units are encouraged.) English units may be used as secondary units (in parentheses). An exception would be the use of English units as identifiers in trade, such as “3.5-inch disk drive”.
- Avoid combining SI and CGS units, such as current in amperes and magnetic field in oersteds. This often leads to confusion because equations do not balance dimensionally. If you must use mixed units, clearly state the units for each quantity that you use in an equation.
- Do not mix complete spellings and abbreviations of units: “Wb/m²” or “webers per square meter”, not “webers/m²”. Spell out units when they appear in text: “. . . a few henries”, not “. . . a few H”.
- Use a zero before decimal points: “0.25”, not “.25”. Use “cm³”, not “cc”).

C. Equations

Number equations consecutively. To make your equations more compact, you may use the solidus (/), the exp function, or appropriate exponents. Italicize Roman symbols for quantities and variables, but not Greek symbols. Use a long dash rather than a hyphen for a minus sign. Punctuate equations with commas or periods when they are part of a sentence, as in:

$$a + b = \gamma \quad (4)$$

Be sure that the symbols in your equation have been defined before or immediately following the equation. Use “(4)”, not “Eq. (4)” or “equation (4)”, except at the beginning of a sentence: “Equation (4) is . . .”

D. \LaTeX -Specific Advice

Please use “soft” (e.g., `\eqref{Eq}`) cross references instead of “hard” references (e.g., (1)). That will make it possible to combine sections, add equations, or change the order of figures or citations without having to go through the file line by line.

Please don’t use the `{eqnarray}` equation environment. Use `{align}` or `{IEEEeqnarray}` instead. The `{eqnarray}` environment leaves unsightly spaces around relation symbols.

Please note that the `{subequations}` environment in \LaTeX will increment the main equation counter even when there are no equation numbers displayed. If you forget that, you might write an article in which the equation numbers skip from (17) to (20), causing the copy editors to wonder if you’ve discovered a new method of counting.

BIB \TeX does not work by magic. It doesn’t get the bibliographic data from thin air but from .bib files. If you use BIB \TeX to produce a bibliography you must send the .bib files.

\LaTeX can’t read your mind. If you assign the same label to a subsubsection and a table, you might find that Table I has been cross referenced as Table IV-B3.

\LaTeX does not have precognitive abilities. If you put a `\label` command before the command that updates the counter it’s supposed to be using, the label will pick up the last counter to be cross referenced instead. In particular, a `\label` command should not go before the caption of a figure or a table.

Do not use `\nonumber` inside the `{array}` environment. It will not stop equation numbers inside `{array}` (there won’t be any anyway) and it might stop a wanted equation number in the surrounding equation.

E. Some Common Mistakes

- The word “data” is plural, not singular.
- The subscript for the permeability of vacuum μ_0 , and other common scientific constants, is zero with subscript formatting, not a lowercase letter “o”.
- In American English, commas, semicolons, periods, question and exclamation marks are located within quotation marks only when a complete thought or name is cited,

such as a title or full quotation. When quotation marks are used, instead of a bold or italic typeface, to highlight a word or phrase, punctuation should appear outside of the quotation marks. A parenthetical phrase or statement at the end of a sentence is punctuated outside of the closing parenthesis (like this). (A parenthetical sentence is punctuated within the parentheses.)

- A graph within a graph is an “inset”, not an “insert”. The word alternatively is preferred to the word “alternately” (unless you really mean something that alternates).
- Do not use the word “essentially” to mean “approximately” or “effectively”.
- In your paper title, if the words “that uses” can accurately replace the word “using”, capitalize the “u”; if not, keep using lower-cased.
- Be aware of the different meanings of the homophones “affect” and “effect”, “complement” and “compliment”, “discreet” and “discrete”, “principal” and “principle”.
- Do not confuse “imply” and “infer”.
- The prefix “non” is not a word; it should be joined to the word it modifies, usually without a hyphen.
- There is no period after the “et” in the Latin abbreviation “et al.”.
- The abbreviation “i.e.” means “that is”, and the abbreviation “e.g.” means “for example”.

An excellent style manual for science writers is [7].

F. Authors and Affiliations

The class file is designed for, but not limited to, six authors. A minimum of one author is required for all conference articles. Author names should be listed starting from left to right and then moving down to the next line. This is the author sequence that will be used in future citations and by indexing services. Names should not be listed in columns nor group by affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization).

G. Identify the Headings

Headings, or heads, are organizational devices that guide the reader through your paper. There are two types: component heads and text heads.

Component heads identify the different components of your paper and are not topically subordinate to each other. Examples include Acknowledgments and References and, for these, the correct style to use is “Heading 5”. Use “figure caption” for your Figure captions, and “table head” for your table title. Run-in heads, such as “Abstract”, will require you to apply a style (in this case, italic) in addition to the style provided by the drop down menu to differentiate the head from the text.

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and,

conversely, if there are not at least two sub-topics, then no subheads should be introduced.

H. Figures and Tables

a) *Positioning Figures and Tables:* Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation “Fig. 1”, even at the beginning of a sentence.

TABLE II
TABLE TYPE STYLES

Table Head	Table Column Head		
	Table column subhead	Subhead	Subhead
copy	More table copy ^a		

^aSample of a Table footnote.



Fig. 1. Example of a figure caption.

Figure Labels: Use 8 point Times New Roman for Figure labels. Use words rather than symbols or abbreviations when writing Figure axis labels to avoid confusing the reader. As an example, write the quantity “Magnetization”, or “Magnetization, M”, not just “M”. If including units in the label, present them within parentheses. Do not label axes only with units. In the example, write “Magnetization (A/m)” or “Magnetization {A[m(1)]}”, not just “A/m”. Do not label axes with a ratio of quantities and units. For example, write “Temperature (K)”, not “Temperature/K”.

ACKNOWLEDGMENT

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

REFERENCES

- Please number citations consecutively within brackets
- b1. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use “Ref. [3]” or “reference [3]” except at the beginning of a sentence: “Reference [3] was the first ...”
- Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors' names; do not use "et al.". Papers that have not been published, even if they have been submitted for publication, should be cited as "unpublished" [4]. Papers that have been accepted for publication should be cited as "in press" [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove the template text from your paper may result in your paper not being published.