Company Names Matching in the Large Patents Dataset

Timofey Medvedev, Alexander Ulanov

HP Laboratories
HPL-2011-90R1

Keyword(s):

Names matching; duplicate detection; clustering; patents

Abstract:

This paper addresses the name matching (duplicate detection) problem in the US patent dataset. It contains more then 400K unique company names spellings. In order to solve the matching problem we choose appropriate string similarity measure and clustering approach and estimate their parameters. Finally we apply them to the whole dataset and estimate the positives and negatives rates.

External Posting Date: July 21, 2011 [Fulltext] Approved for External Publication
Internal Posting Date: July 21, 2011 [Fulltext]

Company Names Matching in the Large Patents Dataset

Timofey Medvedev and Alexander Ulanov

Hewlett-Packard Labs, Information Analytics Lab
1 Artilleriyskaya, St.Petersburg, Russia, 191104
{timofey.medvedev,alexander.ulanov}@hp.com

Abstract. This paper addresses the name matching (duplicate detection) problem in the US patent dataset. It contains more then 400K unique company names spellings. In order to solve the matching problem we choose appropriate string similarity measure and clustering approach and estimate their parameters. Finally we apply them to the whole dataset and estimate the positives and negatives rates.

Keywords: Names matching, duplicate detection, clustering, patents

1 Introduction

Nowadays business needs analytic on structured or semi-structured data. It can be internal or public documents in a specific format. Unfortunately, data in such documents is often compromised by many factors. For example, data entry errors, multiple conventions for words shortening and abbreviation. Also real-world objects can change their names and other properties in time.

In this paper, we investigate the problem of lexical heterogeneity in the collection of US patents [8], [9]. The collection contains more than 4,000,000 of structured documents. We are focused on the assignee field, which contains a name of a company that owns the patent. This field is filled by patent attorney and it may contain misspellings or simply different spellings of the same name. The problem is to group the patents of the same company together and thus make higher recall for patent search.

The paper structures as follows. The next section describes related work. Section 3 presents the problem and tasks. Section 4 discusses experiment setup including datasets generation. Section 5 shows experimental results. Section 6 describes experiments on full dataset. Section 7 contains conclusion and directions for future work.

2 State of the Art

Bilenko et al.[1] describe benchmarks for name matching problem with the use of various datasets. They are using different types of string similarity measures

including static and learnable. Elmagarmid et al.[2] analyze fundamental principles of duplicate detection. They describe in details a lot of string similarity measures. In this paper we focus on the edit based measures since they tend to be the best for the name matching task. We summarized most popular edit distance measures is in Table 1. It contains proposed modification of Levenshtein measure named Levenshtein*.

Table 1. String similarity measures used in experiments.

Measure	**Short description**
Levenshtein	Edit operations (delete, insert, substitute) have cost 1.
Levenshtein*	Upper case edit costs l, upper case to lower case costs $1 + \frac{l}{2}$, where l is an average abbreviation length.
Jaro	Common characters between two strings.
Jaro-Winkler	Jaro modification with higher weight to prefix matches.
Monge-Elkan	Two tokens match if they are equal or if one is the prefix of the another. Similarity is the number of their matching tokens divided by their average number of tokens.
SoftTF-IDF	Two tokens match if similarity of inner-measure more than inner-threshold. Similarity is sum of normalized weights for matching tokens.

Clustering is used when there are more then two instances of the same element with different names. There is a set of papers about efficient clustering of large datasets. McCallum et al.[5] present a technique for clustering the large datasets (*canopies*). Hernandez and Stolfo[3] described Sorted-Neighborhood Method. It can be applied to the sorted data and limit the number of comparisons for each record. Shu et al.[6] proposed new algorithm for the divisive hierarchical clustering based on spectral clustering.

3 Problem Statement and Proposed Approach

We deal with the real-world dataset from US Patents and Trademark Office [8]. It contains more than 4,000,000 full-text patents which are structured documents. We are focused on the assignee field which contains the company name. The main goal of this work is to group the names of the same company. Since there are more than two instances of the same company with different names we will need to do clustering.

We need to choose the best string similarity measure and corresponding threshold as well as clustering algorithm. We can estimate these parameters using existing benchmarks and then apply them on the full collection. We will try different strategies for parameter estimation.

4 Experiments Setup

According to the results presented in [1] we have chosen a set of string similarity measures: *Monge-Elkan, Jaro, Jaro-Winkler, Levenshtein,* and *SoftTF-IDF*. SimMetrics [7] implementation of four first measures were used for our experiments. We implemented Levenshtein* and SoftTF-IDF. Brief information about these measures can be found in the Table 1.

For matching experiment we used *business, kunkel, nybird, parks, scott2,* and *ucd-people* datasets from [1] and five datasets generated by ourselves from patent assignee data. Datasets are listed in the Table 2.

Full-text patents from January 1976 to April 2011 were downloaded from Google Patents. We manage to parse 4191205 from 4401925 document using the developed parser. 440524 unique names were extracted from these patents. We will refer to it as to companies dataset.

For clustering experiment we used *vaUniv* from [1], *cora-ref* from [5] and two datasets from patent companies dataset generated by ourselves. Datasets are listed in the Table 3. We apply agglomerative clustering in the similar way as in [5]. Different strategies are used for computing the distance between clusters, i.e. single link, complete link, and centroid.

Table 2. Matching dataset list

Dataset	Strings	Dataset	Strings
Business	2139	Patents1	341
Kunkel (Bird1)	337	Patents2	280
Nybird (Bird2)	982	Patents3	300
Scott2 (Bird4)	719	Patents4	298
Parks	654	Patents5	298
Ucd-people (UcdFolks)	5332		

Table 3. Clustering dataset list

Dataset	Strings
vaUniv	116
Cora-ref	∼1880
Patents_cl1	203
Patents_cl2	202

We used F1 measure in both experiments to evaluate the results: $F1 = 2 \cdot \frac{recall \cdot precision}{recall + precision}$. During the matching experiments we employed the following formulaes for precision and recall: $recall = \frac{|\{relevant\ pairs\} \cap \{retrieved\ pairs\}|}{|\{relevant\ pairs\}|}$ and $precision = \frac{|\{relevant\ pairs\} \cap \{retrieved\ pairs\}|}{|\{retrieved\ pairs\}|}$. Precision and recall were the following for the clustering experiments: $recall = \frac{TP}{TP+FN}$ and $precision = \frac{TP}{TP+FP}$.

5 Experiments

Table 4 presents a summary of matching experiments results for different measures with respect to the dataset type. Soft-TFIDF line shows average F1 for the different inner-measures (best inner-threshold). Soft-TFIDF$_{max}$ line is the best result among all inner-measures and inner-thresholds, that is Levenshtein*

Table 4. Best F1 results for different measures according to dataset type.

	patent	non-patent	all
Jaro	0.825	0.679	0.740
JaroWinkler	0.869	0.684	0.761
Levenshtein	0.743	0.661	0.695
Levenshtein*	0.824	0.699	0.751
MongeElkan	0.315	0.560	0.458
SoftTFIDF	0.812	0.897	0.861
SoftTFIDF$_{max}$	0.912	0.916	0.914

Table 5. Best F1 results for different measures according to dataset.

	patents_cl1	patents_cl2	vauniv	cora-ref	all
Jaro	0.571	0.671	0.772	0.695	0.677
JaroWinkler	0.665	0.819	0.724	0.499	0.677
Levenshtein	0.604	0.623	0.737	0.902	0.717
Levenshtein*	0.761	0.646	0.766	0.911	0.771
MongeElkan	0.519	0.539	0.763	0.726	0.637
SoftTFIDF	0.878	0.768	0.892	0.8533	0.846
SoftTFIDF$_{max}$	0.904	0.827	0.909	0.915	0.889

Table 6. Best F1 results for different measures according to clustering algorithm.

	single-link	complete-link	centroid	all
Jaro	0.746	0.575	0.711	0.677
JaroWinkler	0.728	0.635	0.668	0.677
Levenshtein	0.727	0.704	0.719	0.717
Levenshtein*	0.781	0.760	0.772	0.771
MongeElkan	0.467	0.661	0.783	0.637
SoftTFIDF	0.812	0.835	0.891	0.846
SoftTFIDF$_{max}$	0.870	0.888	0.908	0.889
(threshold)	(0.64)	(0.42)	(0.5)	

with the similarity threshold 0.9. The best F1 among others are JaroWinkler and Levenshtein*. However Soft-TFIDF delivers the best average results. For the patent domain the average results of Soft-TFIDF were worse than most of the other measures. At the same time Soft-TFIDF$_{max}$ is significantly better. We can conclude that Soft-TFIDF measure demonstrates good results on different input data. It is able to show significantly better results than the other measures if one alpplies some tuning.

Table 6 and 5 presents a summary of clustering experimens results with the use of different similarity measures, cluster similarity strategy, and dataset. Soft-TFIDF line presents average result for different inner-measures (best inner-threshold). Soft-TFIDF$_{max}$ line is the best result among all inner-measures and inner-thresholds, that is Jaro with 0.8 threshold. As one can see, results are similar for different similarity strategies and there is a dependence on the dataset. Soft-TFIDF demonstrates better results in average than other measures.

6 US Patents Assignee Name Matching experiment

The data for these experiment are 440524 unique company names spellings that were extracted from USPTO patents. We could not apply clustering because of the dataset size, so we used a blocking approach similar to the canopies introduced in [5]. We use the first letter of a company name as a cheap distance measure, because we suppose that the same company name starts with the same letter. Overlap coefficient is used as an expensive measure for dividing canopies. Each canopy is clustered using agglomerative (hierarchical) clustering.

SoftTF-IDF with different thresholds is used as a string similarity measure for all the experiments. Jaro is the inner-measure for SoftTF-IDF with the inner-threshold equals to 0.8 (Table 4).

The first round of the final experiments is made with the estimated thresholds (Table 6) and 100 elements are sampled from the result for performance estimation. We apply blocking strategy as mentioned earlier and the apply clustering to each block. We use sinlge link, complete link, and centroid cluster similarity strategy for hierarhical clustering. Estimated precision of the result is poor due to threshold parameters.

We make the second round of experiments. We estimate the parameters on the basis of the sampled data from the first round of experiments. New threshold for the single link strategy is 0.9, for the complete link - 0.85, and 0.78 for the centroid. New experiments take 5, 6 and 19 hours respectively.

The clustering result contain a large number of small clusters. This is an issue for estimating true and false negatives (TN and FN). To deal with this we divided resulting dataset into two parts: *positives* - names in a pair in the same cluster and *negatives* - names in a pair from different clusters. To estimate true positives and false positives (TP and FP) we extracted 50 pairs of names from *positives*. After dividing to TP and FP we estimated the fraction of them using binomial distribution. The same estimation for the *negatives* was rough due to the fact that all extracted pairs were TN. Table 7 shows the estimations

of true and false positives fraction within positives of the full dataset. The rough estimation of false negatives fraction within negatives is presented as well.

Table 7. Estimation of positives and negatives fraction in the full USPTO companies dataset with confidence probability 0.95.

	min % TP	max % of FP	min % of TN	max % of FN	Calculation time	Number of clusters
Single-link	9.8	90.2	94.2	5.8	5 hours	315545
Complete-link	85.2	14.8	94.2	5.8	6 hours	329458
Centroid	82.7	17.3	94.2	5.8	19 hours	295505

7 Conclusion

Our goal was to group company names in the collection of USPTO patents. We picked the best similarity measure and clustering approach based on different benchmarks. We estimated thresholds for similarity measure employed in clustering, applied blocking method to address scalability issue and run clustering on the full companies collection. Error rate was estimated for the whole dataset using results sampling. The best results were shown by the clustering with complete link strategy. Nearly 29% of dataset is duplicated data.

Future work is related with experiments with other blocking strategies. We will explore the ways to estimate the precision and recall of a resulting dataset.

References

1. M. Bilenko, R. Mooney, W. Cohen, P. Ravikumar and S. Fienberg. *Adaptive Name Matching in Information Integration.* IEEE Intelligent Systems, Vol. 18 Is 5, 2003.
2. Ahmed K. Elmagarmid, Panagiotis G. Ipeirotis and Vassilios S. Verykios. *Duplicate Record Detection: A Survey.* IEEE Transactions on Knowledge and Data Engineering, Volume 19 Issue 1, January 2007.
3. Mauricio A. Hernandez, Salvatore J. Stolfo. *Real-world Data is Dirty: Data Cleansing and The Merge/Purge Problem.* Data Mining And Knowledge Discovery, 1998.
4. Christopher D. Manning, Prabhakar Raghavan and Hinrich Schuetze. *Introduction to Information Retrieval.* Cambridge University Press, 2008.
5. Andrew McCallum, Kamal Nigam and Lyle H. Ungar. *Efficient clustering of high-dimensional data sets with application to reference matching.* Knowledge Discovery and Data Mining, p.169-178, 2000.
6. Liangcai Shu, Aiyou Chen, Ming Xiong, Weiyi Meng. *Efficient SPectrAl Neighborhood blocking for entity resolution.* ICDE, p. 1067-1078, 2011.
7. SimMetrics is an open source extensible library of Similarity or Distance Metrics. `http://staffwww.dcs.shef.ac.uk/people/sam.chapman@k-now.co.uk`
8. United States Patent and Trademark Office `http://www.uspto.gov`
9. United States Patent and Trademark Office Bulk Downloads. `http://www.google.com/googlebooks/uspto.html`