Looking for the Best Historical Window for Assessing Semantic Similarity Using Human Literature

Jorge Martinez-Gil
Software Competence Center
Hagenberg
Softwarepark 21
4232, Austria
jorge.martinez-gil@scch.at

Mario Pichler
Software Competence Center
Hagenberg
Softwarepark 21
4232, Austria
mario.pichler@scch.at

Lorena Paoletti
Software Competence Center
Hagenberg
Softwarepark 21
4232, Austria
lorena.paoletti@scch.at

ABSTRACT

We describe the way to get benefit from broad cultural trends through the quantitative analysis of a vast digital book collection representing the digested history of humanity. Our research work has revealed that appropriately comparing the occurrence patterns of words in some periods of human literature can help us to accurately determine the semantic similarity between these words by means of computers without requiring human intervention. Preliminary results seem to be promising.

Keywords

knowledge integration; semantic similarity; culturomics

1. INTRODUCTION

It is widely accepted that the meaning of words evolve over time. However, it is still unclear if word occurrences in human literature along the history can be meaningful in computing word semantic similarity. By semantic similarity measurement we mean the research challenge whereby two terms are assigned a score based on the likeness of their meaning. Automatic measurement of semantic similarity is considered to be of great importance for many computer related fields since a wide variety of techniques. The reason is that textual semantic similarity measures can be used for understanding beyond the literal lexical representation of words and phrases. For example, it is possible to automatically identify that specific terms (e.g., Finance) yields matches on similar terms (e.g., Economics, Economic Affairs, Financial Affairs, etc.). This capability of understanding beyond the lexical representation of words makes semantic similarity methods to be of great importance to the Linked Data community. For example, the ontology alignment problem can be addressed by means of methods of this kind.

The traditional approach for solving this problem has consisted of using manually compiled dictionaries to determine the semantic similarity between terms, but an important problem remains open. There is a gap between dictionaries and the language used by people, the reason is a balance that every dictionary must strike: to be comprehensive enough for being a useful reference but concise enough to be practically used. For this reason, many infrequent words are usually omitted. Therefore, how can we measure semantic similarity in situations where terms are not covered by a dictionary? We think Culturomics could be an answer.

Culturomics consists of collecting and analyzing data from the study of human culture. Michel et al. [8] established this discipline by means of their seminal work where they presented a corpus of digitized texts containing 5.2 million books which represent about a 4 percent of all books ever printed. This study of human culture through digitized books have had a strong positive impact in our core research since its inception. In a previous work [7], the idea of word co-occurrence in human literature for supporting semantic correspondence discovery was explored. Now, we go a step further beyond with a much more complete framework being able to improve our past results. Therefore, the main contributions presented in this work are:

1. We propose to use culturomics for trying to determine the semantic similarity between words[1] by comparing their occurrence pattern in human literature by means of an appropriate statistical analysis.

2. We evaluate a pool of quantitative algorithms for time series comparison to determine what are the most appropriate methods in this context. These algorithms are going to be applied on some statistical transformations which can help to reduce noise.

3. We try to determine what is the best historical time period for computing semantic similarity using human literature.

The rest of this paper is organized as follows: Section 2 describes related approaches that are proposed in the literature. Section 3 describes the key ideas to understand our contribution. Section 4 presents a qualitative evaluation of our method, and finally, we draw conclusions and future lines of research.

[1] We focus in the English language only

2. RELATED WORK

In the past, there have been great efforts in finding new semantic similarity measures mainly due to its fundamental importance in many fields of the modern computer science. The detection of different formulations of the same concept is a key method in a lot of computer-related fields. To name only a few, we can refer to a) clustering [3], service matchmaking [1], web data integration [6], or schema matching [2] rely on a good performance when determining the meaning of data.

If we focus on the field of semantic change, we can see how authors define it as a change of one or more meanings of the word in time. Developing automatic methods for identifying changes in word meaning can therefore be useful for both theoretical linguistics and a variety of applications which depend on lexical information. Some works have explored this path, for instance [10] investigated the significant changes in the distribution of terms in the Google N-gram corpus and their relationships with emotion words or [5] who presented an approach for automatic detection of semantic change of words based on distributional similarity models. Our approach is different in the sense we compute semantic similarity using a specific historical window.

3. CONTRIBUTION

Our contribution is an analysis of books published along the history. The aim is to build novel measures which can determine the semantic similarity of words in an automatic way. The main reason for preferring this paradigm rather than a traditional approach based on dictionaries is obvious; according to the book library digitized by Google[2], the number of words in the English lexicon is currently above a million. Therefore, there are more words from the data sets we are using than in any dictionary. For instance, the Webster's Dictionary[3], lists much less than 400,000 single-word word forms currently [8].

We have chosen ten well-known algorithms for time series comparison. This pool includes distance measures (Euclidean, Chebyshev, Jaccard, and Manhattan) , similarity measures (Cosine, Dynamic Time Warping, Roberts, and Ruzicka), and correlation coefficients (Pearson and Spearman's correlation) [4]. We provide a brief description for each of these algorithms listed in alphabetical order below. We consider that the pair x and y are the time series representation for each of the words to be compared.

1. *Cosine similarity* is a measure between two time series which determines the cosine of the angle between them.

$$sim(x,y) = \frac{\sum_{i=1}^{n} x_i \cdot y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}} \tag{1}$$

2. *Euclidean distance* computes the euclidean distance between each two points along the time series.

$$sim(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{2}$$

3. *Chebyshev distance* computes the greatest difference along any two points in the time series.

$$sim(x,y) = max_{i=1}^{n} |x_i - y_i| \tag{3}$$

4. *Dynamic Time Warping* uses a dynamic programming technique to determine the best alignment that will produce the optimal distance.

$$sim(x,y) = \sum_{i=1,k=1}^{n,m} |x_{ik} - y_{ik}| \tag{4}$$

5. *Jaccard distance* measures the similarity of two sets by comparing the size of the overlapping points against the size of the two time series.

$$sim(x,y) = \frac{\sum_{i=1}^{n} (x_i \wedge y_i)}{\sum_{i=1}^{n} (x_i \vee y_i)} \tag{5}$$

6. *Manhattan distance* computes the sum of the absolute values of the differences between the corresponding points from the time series.

$$sim(x,y) = \sum_{i=1}^{n} |x_i - y_i| \tag{6}$$

7. *Pearson Correlation* determines the ratio between the covariance and the standard deviation of two time series.

$$sim(x,y) = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n} (y_i - \bar{y})^2}} \tag{7}$$

8. *Roberts similarity* examines the relation between the sum of each two corresponding points within the min and max of them.

$$sim(x,y) = \frac{\sum_{i=1}^{n} (x_i + y_i) \cdot \frac{min\{x_i,y_i\}}{max\{x_i,y_i\}}}{\sum_{i=1}^{n} (x_i + y_i)} \tag{8}$$

9. *Ruzicka similarity* tries to find the difference between each of two corresponding pairs divided by the maximum for each case.

$$sim(x,y) = \frac{\sum_{i=1}^{n} min(x_i, y_i)}{\sum_{i=1}^{n} max(x_i, y_i)} \tag{9}$$

10. *Spearman's rank correlation* is a statistical measure that tries to find if there is a monotonic relationship between the two time series.

$$sim(x,y) = 1 - \frac{6\sum (x_i - y_i)^2}{N(N^2 - 1)} \tag{10}$$

Therefore, our contribution is a framework where the problem is addressed using different perspectives: a) algorithms for comparing time series similarity, b) statistical transformations of time series using reduction, baseline removal, rescaling and smoothing techniques, and c) looking for the most appropriate time window, thus, the range of years which helps us to perform the most accurate predictions.

[2] http://books.google.com/ngrams
[3] http://www.merriam-webster.com

3.1 Working with statistical transformations

Working with time series has a number of problems since two similar time series can present the same pattern but different occurrence volumes. This can be solved by means of normalization techniques. However, there are some algorithms where normalization has not any kind of effect, for instance when using Cosine Distance which tries to measure the angle between the two vectors of numeric values.

3.1.1 *Smoothing of the original time series*

Smoothing a time series consists of creating an approximating function to capture important patterns, while leaving out noise or other disturbing phenomena. Therefore, smoothing is a widely used technique for reducing of canceling the effect due to random variations. This technique, when properly applied, reveals more clearly the underlying trend of the time series. We want to run the algorithms in smoothed data because this kind of technique can help us to obtain cleaner time series and, therefore, results are going to reflect trends more clearly.

3.2 Looking for the best time window

Methods presented until now can give us some advice about what direction should be explored. However, these results are far from being considered optimal. One of the main reasons is that we have only focused in a fixed time period. In order to overcome this limitation, we have designed an algorithm for trying to capture the optimal time window for solving the Miller-Charles benchmark data set.

4. EVALUATION

We report our results using the 1-gram data set offered by Google[4]. The data is in the range between 1800 and 2000. The reason is that there are not enough books before 1800 to reliably quantify many of the queries from the data sets we are using. On the other hand, after year 2000, quality of the corpus is lower since it is subject to many changes. Results are obtained according Miller-Charles data set [9]. The rationale behind this way to evaluate quality is that the results obtained by means of artificial techniques may be compared to human judgments.

4.1 Evaluation with classic algorithms

Table 1 shows the results over the raw data. The Euclidean distance presents the best performance. However, the scores obtained are very low. This is the reason we propose to apply some statistical transformations.

4.2 Statistical transformations over the time series

Table 2 shows the results after normalizing the data sets within the real interval [0, 1]. This means that all the occurrences of the terms along the history have to be compressed in this real interval, where 0 means no occurrences and 1 means the maximum number of occurrences.

Noise on time series may be due to varying or bad baselines. The baselines in a time series can be fitted to and removed by subtracting from each value the average mean of the time series. Table 3 shows the results after removing the baseline for the data sets.

Algorithm	Score
Cosine	0.28
Chebyshev	0.23
DTW	0.21
Euclidean	0.30
Jaccard	0.09
Manhattan	0.29
Pearson	0.28
Roberts	0.10
Ruzicka	0.11
Spearman	0.08

Table 1: Results working with raw data.

Algorithm	Score
Cosine	0.28
Chebyshev	0.29
DTW	0.35
Euclidean	0.32
Jaccard	nosense
Manhattan	0.26
Pearson	0.28
Roberts	0.23
Ruzicka	0.24
Spearman	0.36

Table 2: Results after normalizing data sets in [0,1].

Rescaling a time series is a method which consists of dividing the range of the values exhibited in time series by the standard deviation of the values. Table 4 shows the results obtained after rescaling original data.

4.2.1 *Smoothing of the time series*

One of the best-known smoothing methods is the Moving Average (MA) technique which takes a certain number of past periods and add them together; then it divides them by the number of periods. Table 5 shows the results when using smoothed time series using MA for the periods 5, 10, 20 and 50 years respectively. Another popular smoothing method is called Exponential Moving Average (EMA) technique which applies more weight to recent data. The weighting applied to the most recent data depends on the number of periods. Table 6 shows the results when using smoothed time series using EMA for the periods 5, 10, 20 and 50 years.

4.3 Best historical window

Until now, we have only focused in the fixed time period between 1800 and 2000. In order to overcome this limitation, we have designed an algorithm for trying to capture the optimal time window for solving the Miller-Charles benchmark data set. The algorithm we have designed is able to test every possible configuration for the time windows (with a minimum size of 2 years), computational algorithm used and statistical transformation for data. This means we have automatically tested 2,412,000 different configurations (20,100 different windows over 12 different statistical transformations using 10 different algorithms). The best results we have achieved are summarized in Table 7. We can see that using the Pearson correlation coefficient between the years 1935 and 1942 using raw data or between 1806 and 1820 over a moving average of five years allows us to solve the

[4] https://books.google.com/ngrams

Algorithm	Score
Cosine	0.26
Chebyshev	0.22
DTW	0.15
Euclidean	0.31
Jaccard	0.09
Manhattan	0.31
Pearson	0.28
Roberts	0.09
Ruzicka	0.15
Spearman	0.07

Table 3: Results after baseline removal.

Algorithm	Score
Cosine	0.28
Chebyshev	0.41
DTW	0.35
Euclidean	0.30
Jaccard	0.37
Manhattan	0.22
Pearson	0.28
Roberts	0.28
Ruzicka	0.26
Spearman	0.37

Table 4: Results after rescaling data.

Miller-Charles benchmark data set [9] with a high accuracy. This means that our hypothesis stating that an appropriate combination of: algorithms, statistical transformation and time windows could lead to positive results is confirmed.

5. CONCLUSIONS

We have described how we have perform a quantitative analysis of a vast digital book collection representing a significant sample of the history of literature to solve problems related to the semantic similarity. In fact, we have shown that appropriately choosing a combination of quantitative algorithms for comparing time series representing the occurrence patterns, some statistical transformations on source data which can help to reduce noise, and the election of a correct time window can provide very accurate results when measuring semantic similarity between single words.

Algorithm	M(5)	M(10)	M(20)	M(50)
Cosine	0.27	0.25	0.24	0.25
Chebyshev	0.27	0.27	0.25	0.22
DTW	0.22	0.24	0.24	0.25
Euclidean	0.30	0.29	0.29	0.28
Jaccard	0.20	0.21	0.19	0.31
Manhattan	0.29	0.29	0.28	0.27
Pearson	0.23	0.21	0.18	0.15
Roberts	0.10	0.10	0.10	0.11
Ruzicka	0.11	0.11	0.11	0.12
Spearman	0.08	0.08	0.08	0.09

Table 5: Results after smoothing time series using moving averages (5, 10, 20 and 50 years).

Algorithm	E(5)	E(10)	E(20)	E(50)
Cosine	0.27	0.26	0.25	0.26
Chebyshev	0.26	0.26	0.25	0.22
DTW	0.18	0.24	0.26	0.26
Euclidean	0.30	0.29	0.29	0.29
Jaccard	0.14	0.21	0.18	0.21
Manhattan	0.29	0.29	0.28	0.27
Pearson	0.23	0.21	0.18	0.06
Roberts	0.10	0.10	0.10	0.11
Ruzicka	0.11	0.11	0.11	0.12
Spearman	0.08	0.08	0.08	0.10

Table 6: Results after smoothing data using exponential moving averages (5, 10, 20 and 50 years).

Time Windows	Algorithm	Data	Score
1935-1942	Pearson	Raw Data	0.67
1806-1820	Pearson	MA(50)	0.67
1940-1942	Pearson	EMA(5)	0.65

Table 7: Best time windows for solving the Miller-Charles benchmark data set using culturomics.

Acknowledgments

We thank the reviewers for their useful comments. This work has been funded by ACEPROM (Proj. Nr. 841284) funded by the Austrian Research Promotion Agency (FFG).

6. REFERENCES

[1] Bianchini, D., De Antonellis, V., Melchiori, M. Flexible Semantic-Based Service Matchmaking and Discovery. World Wide Web 11(2): 227-251 (2008).

[2] Castano, S., Ferrara, A., Montanelli, S., Lorusso, D.: Instance Matching for Ontology Population. SEBD 2008: 121-132.

[3] De Virgilio, R., Cappellari, P., Miscione, M. Cluster-Based Exploration for Effective Keyword Search over Semantic Datasets. ER 2009: 205-218.

[4] Deza, M.M., Deza, E. Encyclopedia of Distances. 2013. Springer.

[5] Gulordava, K., Baroni, M. A distributional similarity approach to the detection of semantic change in the Google Books Ngram corpus. GEMS 2011: 67-71.

[6] Martinez-Gil, J., Aldana-Montes, J.F. Semantic similarity measurement using historical google search patterns. Inf. Syst. Frontiers 15(3): 399-410 (2013).

[7] Martinez-Gil, J., Picher, M. Analysis of word co-occurrence in human literature for supporting semantic correspondence discovery. I-KNOW 2014: 1:1-1:7.

[8] Michel, J.B., Shen, Y., Aiden, A., Veres, A., Gray, M., Pickett, J., Hoiberg, D., Clancy, D., Norvig, P., Orwant, J., Pinker, S., Nowak, M., and Aiden, E. Quantitative analysis of culture using millions of digitized books, Science 331(6014): 176-182 (2011).

[9] Miller, G.A., Charles W.G. Contextual correlates of semantic similarity. Language and Cognitive Processes, 6(1):1-28 (1991).

[10] Popescu, O., Strapparava, C. Behind the Times: Detecting Epoch Changes using Large Corpora. IJCNLP 2013: 347-355.