2019 IEEE Fifth International Conference on Big Data Computing Service and Applications (BigDataService)

SOREC: A Semantic Content-based Recommendation System for Parsimonious Sociology Theory Construction

Mingzhe Du
Dept. of Computer Science and Engineering
University of South Carolina
Columbia, USA
dum@email.sc.edu

Jose M. Vidal
Dept. of Computer Science and Engineering
University of South Carolina
Columbia, USA
vidal@sc.edu

Barry Markovsky
Dept. of Sociology
University of South Carolina
Columbia, USA
barry@sc.edu

***Abstract*—Theory construction is the process of formulating scientific theories with reference to explicit logical and semantic criteria. Definitions and associated terms are essential components of the theory, in which parsimony is a crucial criterion for theory evaluation. The present work offers a novel semantic content-based recommendation system with supervised machine learning model for theoretical parsimony evaluation by checking the semantic consistency of definitions while constructing theories. Specifically, we evaluate the XGBoost tree-based classifier with the combination of 15 low-level features and 11 high-level features on our dataset. A sociologist annotated in-house dataset consisting of $2,235$ definition pairs drawn from the sociological literature is used for evaluating the proposed methods. The experiment results showed that the proposed system achieves 86.16% accuracy, 84.42% F-measure and 86% precision in suggesting semantically related sociological definitions.**

***Index Terms*—semantic analysis, content-based filtering, recommendation system, sociology, theory construction**

I. Introduction

In the social sciences, theory construction is the research process of building theories, where theories are used to explain and predict observed phenomena in the natural world [1], [2]. Terms represent concepts or ideas in a theory, and their meanings are explicated in the definitions [1], [2]. The parsimony principle – as exemplified by the notion of Occam's Razor – using relatively few definitions (terms) for theory construction, is an important criterion for evaluating the quality of theories [1]–[4]. Conventional methods for parsimony analysis in social science theory construction are based on abductive heuristic approaches, which are determined by the human, and the results are often lack of coherence and logical integrity [1], [4]. Little effort has been paid to encouraging a more scientific approach through the use of statistical models.

In this paper, we propose a novel approach using content-based recommendation system (CBRS) to promote the parsimony of a theory [7]–[9]. Specifically, during the process of theory construction, the CBRS calculates the semantic similarities [17] of user-entered definitions, then provides suggestions to eliminate redundancies. This ensures the semantically similar definitions for different terms are made to converge into one definition for a single term.

With the explosion of big data and model-based CBRS [11], [12], [23]–[25], there are multiple approaches to tackle this problem. One of them is a semantic ontology-based approach, which use of WordNet [17]–[19] in enhancing semantic-based analysis where hierarchies of concepts are built to capture conceptual relations between words and sentences. This approach performs well on the general domain, but many sociological terms are utilized and/or defined differently from generic English.

Another promising approach is based on Latent Semantic Analysis, which originates from the principle that words used in the same context tend to have similar meanings. Semantic relatedness is discovered through matrix factorization [31]. This approach performs well on various CBRS [24], [25], and is considered as the baseline method.

In recent years, word embeddings and sentence embeddings have produced high-quality representations for words and sentences on a wide spectrum of natural language understanding applications [20], [30]. Especially, deep neural language models have demonstrated the efficacy by using pre-trained language models followed by fine-tuning dataset and achieved state-of-the-art results in semantic similarity related tasks [20]. Considering the excellent performance on representing the semantic similarities, the definitions in our study are embedded with Transformer in [20].

The proposed SOciology RECommender (SOREC) is designed to check the semantic consistency of sociological definitions, which consists of three components: data pre-processing, feature extraction, and definition recommendation. For data pre-processing, the definitions were extracted from a collection of sociological books, then annotated by sociologists. Prior to feature extraction, definitions were tokenized, stop words were removed, and all words were converted to the lower case. For feature extraction, we exploit 15 low-level features from the basic properties of definitions and the edit distance, 11 high-level features from the embedding-based distance metrics. For definition training and recommendation, we adopt XGBoost [35] on the feature representations extracted from 26 features to predict the definition similarity. To the

978-1-7281-0059-3/19/$31.00 ©2019 IEEE
DOI 10.1109/BigDataService.2019.00025

best of our knowledge, our work is the first attempt to use machine learning-based CBRS on promoting the parsimony for sociology theory construction.

The main contributions of this paper are as follows: (1) We proposed a novel semantic CBRS which adapted specifically for our domain of interest. (2) We compared the importance of the feature sets and analyzed the impact of each feature set. (3) Additionally, we presented a manual annotation benchmark dataset for the sociological definition of similarity estimates, which can be used for training and evaluation in future research in this field. The results of this study can be further applied to the theory construction of psychology, criminology, and other social sciences.

The rest of this paper is organized as follows. Section II briefly reviews the related work. Section III describes the proposed machine learning model-based semantic analysis method. Evaluation results and discussions are presented in Section IV. The conclusion is in the last section.

II. RELATED WORKS

A. Parsimonious Theory Construction

A successful theory is one that, when applied to specific empirical cases, describes relationships among phenomena, and explains and predicts the occurrence of certain events. Good scientific theories include four essential components: terms, statements, arguments and scope conditions [1]–[3]. Terms are used to build statements; statements are used to build arguments; arguments apply under a set of scope conditions. Terms in a theory are carefully chosen by the theorist to convey ideas or concepts. Definitions are themselves made of other terms whose meaning must be clear [1], [2], [4].

Wikitheoria is a fully functional knowledge aggregation web framework hosted on the Google Cloud Platform. It is built for modularized theory construction in the social sciences. SOREC is one of Wikitheoria's sub-systems built for the purpose of facilitating the construction of parsimonious theories. Parsimony favors the use of relatively few definitions (terms), rather than creating new ones, when the user goes to add the new definitions.

For example, consider the following two definitions for the term "ambivalence" extracted from the Blackwell Encyclopedia of Sociology.

- D1: the presence in one person at the same time of two competing or conflicting emotions or attitudes
- D2: simultaneous conflicting feels toward a person or an object

If D1 were in the theory already and the user entered D2 as a new definition, the semantic recommendation system would determine whether D2 is in the theory or is similar to D1. If either is the case, the system recommends using D1. As such, the system helps safeguard against redundancy and fosters the more parsimonious theories.

B. Recommendation System

A recommender system is defined as "A system that has as its main task choosing certain objects that meet the requirements of users, where each of these objects are stored in a computer system and characterized by a set of attributes." [40] It helps users to quickly discover the information they need in a specific context through information filtering. Most of these recommendations are implemented in three filtering methods: collaborative filtering, content-based filtering, and hybrid filtering [5]–[10].

The collaborative filtering method learns from the users' past activities and uses their common behavior patterns to make recommendations that the user may be interested in [5], [6]. Content-Based filtering focuses on the characteristics of the recommended item. For example, when searching for a similar pre-defined definition from the lexicon, the recommendation output is based on its syntactic and semantic relatedness [7]–[9], [11]. Hybrid filtering is a combination of CF and CBF [10]. According to CBF's prior knowledge, the main source of information used in content-based filtering systems is text fragments [7]. A set of encoding methods, typically TF-IDF are used to present the definitions [37]. However, in our study, a number of semantically equivalent sociological terminologies are used to construct definitions. The traditional IR methods work fine on the general domain but fail to capture the semantic similarity in the sociological domain. Therefore, natural language processing and machine learning based models are currently used to analyze, classify or measure the latent semantic similarity to support the CBF.

C. Semantic Analysis

In recent years, word embedding and sentence embedding techniques have gained substantial improvement in natural language understanding [20], [30]. Mikolov et al. have illustrated the effectiveness of neural word representations for similarities and other neural language processing algorithms. For word similarity measurements, 63.7% and 65.6% accuracy were achieved by using Continuous Bag of Words and Skip-gram respectively with the corpus of 1.6 billion words [30]. In 2018, Cer et al. demonstrated the good performance of Transformer embedding on semantic similarity tasks [20].

Many efforts have been made to develop semantic textual similarity datasets and related models [13], [21], [22]. For example, the SemEval Semantic Textual Similarity (STS) challenges have been organized for over six years. These challenges greatly accelerated semantic textual research. Manually annotated datasets given by STS empowered the improvement and examination of various methods for semantic similarity estimation. Many supervised learning models were shown to be well performed for semantic recommendation [14]–[16]. Diverse features such as WordNet-based sentence distance calculation and distributed word embedding representations were shown to perform well for comparable semantic text similarity computations on the general domain [17]–[19], [39].

III. PROPOSED APPROACH

In this section, we specify the details of our data and provide the details of the methodology used to perform our analysis.

The general architecture of our proposed recommendation process is illustrated in Fig. 1.

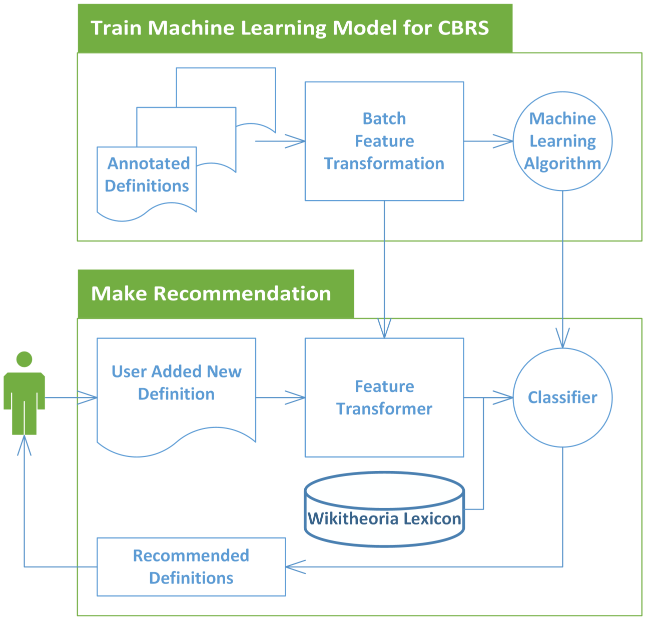
Train Machine Learning Model for CBRS
Annotated Definitions
Batch Feature Transformation
Machine Learning Algorithm
Make Recommendation
User Added New Definition
Feature Transformer
Classifier
Wikitheoria Lexicon
Recommended Definitions

Fig. 1. SOREC Recomendation Architecture

A. Dataset

As far as we know, there is neither a similar dataset we could use nor a similar recommender system or published research on pairwise sociological definition semantic similarity computation. We therefore created a benchmark dataset for this purpose. The dataset includes 2235 definition pairs, including 959 positive samples and 1276 negative samples. This dataset was compiled from the glossaries of a broad range of sociological books[1]. These glossaries were used to generate pairwise comparisons of definitions offered for single terms, then evaluated by sociologists who judged their similarity with scores of 0 (different concept) and 1 (same concept). The SOREC dataset of sentence pairs is publicly available at http://github.com/mzdu/SOREC.

B. Data Preprocessing

Pre-processing is an essential step to improve the accuracy of model prediction. It can both reduce the time complexity for model training and accelerate the system response for online model predictions. In our study, pre-processing methods include tokenizing the definitions, removing the stop word and converting all words to lower case were applied to the definitions before applying the features analysis. We evaluated definition similarities based on the character level and term level briefly described in the following subsections on the basis of an annotated dataset. The TreeBank tokenizer implemented in the NLTK toolkit was used to convert a definition sentence to a list of tokens [32].

C. Feature Extraction with Definitions

Textual similarity metrics detect similarities between the two definitions. These are then used as features for our machine learning models. Zobel and Moffat analyzed a range of similarity measures in information retrieval and found there to be no one-size-fits-all metric, i.e., no metric that consistently worked better than others [38]. To optimize our model, we spent considerable effort on feature engineering, experimented with the different combination of feature sets and obtained the best result with the features described below.

1) Descriptive Feature Set: The descriptive feature set includes basic properties of definition sentences. It presents observations about the characteristics of definitions. In this feature set, we calculated their lengths, length difference, character counts (excluding spaces), word counts, and words in common.

2) Tokenized Feature Set: In general, replaced words, inserted words, and missed words frequently occur in similar definitions. Tokenized feature set calculates the edit distance between one definition and another, i.e., the minimum number of operations that it would take to transform one definition into the other. We first processed definitions as two sets of sorted/unsorted token lists, then evaluated the similarity of pairwise token sets by calculating the minimum number of primitive operations including insertion, deletion, substitution, or copying of a character required to convert one string into the exact match of the other. Specifically, we calculated the normalized Levenshtein distances [29] of pairwise tokens to generate the features based on the overlap ratio of unsorted token sets, the overlap ratio of sorted token sets, the overlap ratio of an unsorted partial token set and the overlap ratio of a sorted partial token set. The output ratio is on a 0 to 100 scale.

3) Embedding Feature Set: Tokenized features measure similarity based on exact matches between isolated words, but not their semantic meanings in context. The Universal Sentence Encoder [20] mixed an unsupervised task using a large corpus together showed significant improvement by leveraging the Transformer architecture, which is based on the attention mechanism. We trained our definitions with Tensorflow Hub[2] transformer encoders. Each definition was transformed into a 512-dimensional sentence vector. With the Transformer encoded embedding output, we computed the distance of two definition vectors (u and v) with the kernel functions shown below.

Cosine Distance [28] between two vectors u and v is defined as

$$sim_1 = 1 - \frac{u \cdot v}{\|u\|_2 \, \|v\|_2} \tag{1}$$

Manhattan Distance [28] computes the distance between two vectors u and v by summing the differences of their

[1] The list of books is available upon request.

[2] Avaialble at http://www.tensorflow.org/hub/modules/google/universal-sentence-encoder-large/3

corresponding components, which is defined as

$$sim_2 = \sum_i |u_i - v_i| \qquad (2)$$

Jaccard Distance [27] proposed by Jaccard and Needham measures the dissimilarity between two vectors u and v, is defined as

$$sim_3 = \frac{u \cdot v}{|u|^2 + |v|^2 - u \cdot v} \qquad (3)$$

Canberra Distance [26] between two vectors is defined as

$$sim_4 = \sum_i \frac{|u_i - v_i|}{|u_i| + |v_i|} \qquad (4)$$

Euclidean Distance [33] between 1-D arrays u and v is defined as

$$sim_5 = \|u - v\|_2 \qquad (5)$$

Minkowski Distance [28] between 1-D arrays u and v is defined as

$$sim_6 = \|u - v\|_p = (\sum |u_i - v_i|^{p}))^{1-p} \qquad (6)$$

Bray-Curtis Distance [34] is defined as

$$sim_7 = \sum |u_i - v_i| \,/\, \sum |u_i + v_i| \qquad (7)$$

Skewness and Kurtosis [41] are the parameters used to measure the symmetry of the dataset and the weight of the tail compared to the normal distribution. Skewness is a measure of the symmetry. If the distribution or dataset appears to be the same as the left and right sides of the center point, it is symmetrical. Kurtosis is a measure of tail thickness, i.e., distribution with high kurtosis often has a heavy tail or outliers. Datasets with low kurtosis tend to have a light tail or outliers.

D. Model

With the features we created in Section C, our goal was to create a relevance model that would accurately predict if a user added new definition is semantically similar to an existing definition in the theory. Our model is built with XGBoost, proposed by Chen and Gestrin in 2016 [35], an optimized distributed gradient boosting library. Gradient boosting is a popular technique that can solve complex regression or classification task by producing and combining a number of weaker and smaller prediction models in the form of decision trees. The model is built in stages and generalized by optimizing a differential loss function.

As a result, gradient boosting combines a number of weak learners into a single, strong learner on an interactive basis. In contrast to linear classifiers (such as logistic regression), decision tree models also can capture non-linear relationships in data. We estimate the best hyperparameter settings for each model using a grid search with 10-fold cross-validation on the training set [36]. Carefully tuning the tree-related hyperparameters (such as the maximum depth of a tree) results in the largest increase of cross-validation F1 score and accuracy. Tuning the learning rate is effective in preventing overfitting on the training data. Using a large number of estimators results in the best performance overall, with training time increasing proportionally.

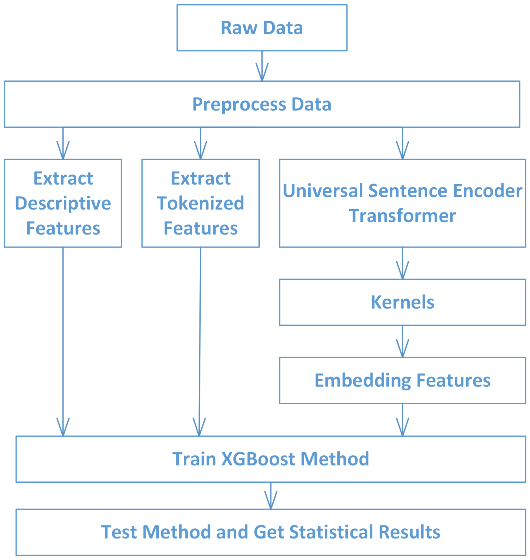
Raw Data
Preprocess Data
Extract Descriptive Features
Extract Tokenized Features
Universal Sentence Encoder Transformer
Kernels
Embedding Features
Train XGBoost Method
Test Method and Get Statistical Results

Fig. 2. Prediction Workflow

In our experiment, we chose tree booster in XGBoost as described in this section for all the feature representations' evaluation, in which max tree-depth was 15, step size shrinkage was 0.1, n estimators was 800 and minimum loss reduction was 1.0.

As shown in Fig. 2, first, descriptive features and tokenized features were extracted from the definition pairs. These two feature sets were used to directly calculate the similarity of two definitions with respect to basic properties and edit distance. Then, these features adopted kernel-based Transformer sentence encoding to calculate the similarity of two definitions. All these similarity scores were concatenated as features and evaluated in the machine learning XGBoost model.

E. Evaluation

For this study, the XGBoost model evaluation was performed using 10-fold cross-validation over 2235 sociological definition pairs, including 959 positive samples and 1276 negative samples. Each fold contained 1811 definitions pairs for training, 200 pairs for validation and 224 pairs for testing.

The final result for the supervised semantic content-based filtering was calculated by averaging the results of each fold. The quality and correctness of the proposed method is evaluated as 1) True positive (TP), the number of correct predictions on same concept; 2) True negative (TN), the number of correct predictions on different concept; 3) False positive (FP), the number of wrong predictions on same concept; 4) False negative (FN), the number of wrong predictions on different concept. The precision (8), recall (9), F-measure (10) and accuracy (11) were used to evaluate the recommendation system.

$$Precision = \frac{TP}{TP + FP} \qquad (8)$$

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

$$F - measure = 2 * \frac{Precision}{Precision + Recall} \quad (10)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (11)$$

IV. Results and Discussion

In this section, we validate the effectiveness of our proposed method from two experiments. First, we evaluate the proposed method in terms of classical metrics of precision, recall, F-measure, and accuracy to justify the usefulness of each feature set in our method. Second, we break down the results and compare the improvement with normalized confusion matrix.

A. Precision, Recall, F-measure and Accuracy

In this study, our feature representations are extracted from three different categories: descriptive feature set (DF), token feature set (TF) and embedding feature set (EF). TABLE I shows results for corresponding feature categories that were used as model input. To evaluate the effectiveness of different categories, we performed several experiments on different combinations of feature categories. From the results, both DF and TF obtained moderate precision, recall, and accuracy. Comparing with EF or DF, the range of increase in precision, recall, and accuracy is 5% to 12%. Between three categories, our experiments show that the EF contributed more to the performance compared with other feature sets. When concatenating EF with DF or TF, the increase in precision varies from 5% to 15%. This is an expected result because embedding-based features enclose the transfer learning from billion words corpus and the measurement from multiple dimensional spaces. To capture the semantics, EF shows that it is a promising method for representing definitions as vectors while capturing semantics. TABLE I shows a significant improvement in precision, recall, F-measure, and accuracy when DF, TF, and EF are concatenated.

The experimental results indicate that the model with a combination of three feature categories outperforms the individual performance of each category and the combination of any two categories, which means that these feature sets complement each other. Although the embedding-based feature set obtained the strongest performance on precision and accuracy, the combination of three categories by a supervised algorithm had the best performance on all metrics. The proposed supervised semantic analysis model achieves the best precision of 86%, the best F-measure of 84.42% and the best accuracy of 86.16%.

We also compared Transformer-embedded definitions with average pooling Google News pre-trained embeddings on the same XGBoost model described in the previous section. As shown in TABLE II, Transformer outperforms Word2Vec by 2% on all metrics.

TABLE I
Results on test data with 10-fold cross validation

Model	Precision	Recall	F-measure	Accuracy
TF-IDF + W2V + SVD	0.68	0.67	0.5780	0.6741
DF	0.70	0.69	0.5868	0.6919
TF	0.77	0.76	0.6936	0.7633
EF	0.84	0.83	0.8140	0.8348
DF-TF	0.76	0.76	0.7066	0.7589
DF-EF	0.85	0.85	0.8265	0.8482
TF-EF	0.84	0.84	0.8275	0.8437
DF-TF-EF	0.86	0.86	0.8442	0.8616

TABLE II
Results with Word2Vec and Transformer

Embedding	Precision	Recall	F-measure	Accuracy
Word2Vec	0.84	0.84	0.8258	0.8437
Transformer	0.86	0.86	0.8442	0.8616

B. Usefulness of Each Feature Category

In the previous section, we saw a steady improvement when three categories were gradually added to the prediction model. For the rest of this subsection, we conducted a three-step experiment to justify the usefulness of each feature set and to inspect the effectiveness of each category.

DF considers the length related descriptions of definitions. As shown in Fig. 3 and TABLE I, the TP is 49% and TN is 85%. The result is expected, as in the general domain, similar definitions tend to be of similar length. When definitions substantially differ in length, the model tends to predict dissimilarity. But in sociology, the "same concept" could be defined with one short sentence or multiple sentences if terminologies are densely used to support definitions. In our dataset, the average length of a definition is 17.87, and the average length difference between pair definitions is 9.67. The definition pairs with high relative length difference tend to be harder to predict correctly for the "same concept." With DF representation, the model achieved an average accuracy of 69.19% and F-measure of 57.8%.

Next, we validate the usefulness of TF. For the tokenized feature set, we calculated the edit distance between one definition and another. As shown in Fig. 4 and TABLE I, with DF and TF, the recommendation quality for correctly recommending the "same concept" definitions have been improved by 16%. The F-measure is 11.97% higher than the previous step, and the prediction accuracy achieves 75.89%.

With the added EF, from Fig. 5 and TABLE I, we see a 19% improvement on TP and 4% improvement on TN. 84% of "same concept" definitions are correctly predicted, and 88% of "different concept" definition pairs are also correctly predicted. This indicates the sociological semantic relatedness could be well represented by calculating the embedding distance from multiple dimensions. The F-measure is improved by 1.67%, and the best prediction accuracy achieves 86.16%.

As shown in Fig 6, the baseline recommendation sys-

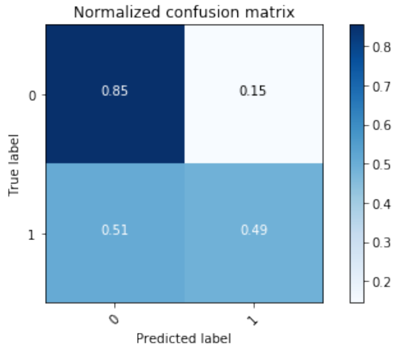

Fig. 3. Prediting with DF

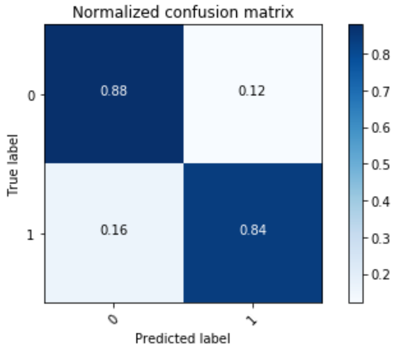

Fig. 5. Prediting with DF, TF and EF

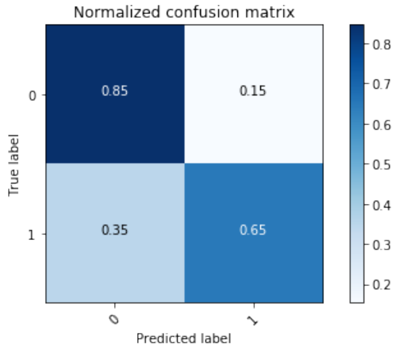

Fig. 4. Prediting with DF and TF

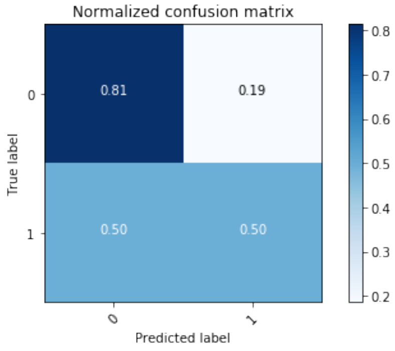

Fig. 6. Prediting with TF-IDF, Word2Vec and SVD

tem is based on the TF-IDF, Word2Vec and Singular Value Decomposition (SVD) model. To evaluate the performance of the baseline model, we encoded definitions with TF-IDF Word2Vec, applied SVD, then fed them to the same XGBoost model we constructed in Section III. From Fig. 6 and TABLE I, this model shows a weak ability to predict “same concept” definitions with TP of 50%. Comparing with the baseline model, the proposed semantic content-based recommender system improves the lexicon semantic similarity recommendation and achieves the best performance.

V. Conclusion

In this study, we proposed a novel semantic content-based recommender system for sociological theory construction. To the best of our knowledge, there is neither a similar recommender system nor published research on the semantic evaluation of sociological definitions. We demonstrated the need for a semantic recommender for semantic level analysis and the effectiveness of our proposed approach to understand the semantic similarity of terminologies and definitions in the sociological domain. Another important contribution of this study is to provide a solid baseline as well as a sociologists-annotated benchmark dataset for future studies in this research area.

Our results revealed that the descriptive features, the edit distance based tokenized features and the kernel function based embedding features complement each other. Particularly, the high-level features consist of the embedding vector distances calculated from unsupervised kernel functions with the help of an XGBoost model increased the overall performance of the recommender system.

The sociologists annotated dataset is publicly available on Github. The proposed CBRS is deployed and serving as a part of Wikitheoria platform. Theory construction is a common research process in a lot of human science-related disciplines, such as Psychology, Criminology, etc. Our sociology-domain specific semantic sentence-level similarity measures can also be applied to various applications for parsimonious theory construction in these disciplines.

VI. Acknowledgments

The authors are supported by the SES-1123040 and IIS-1551458 grants from the National Science Foundation. We would like to thank Jake Frederick, Nicolas Harder for annotations and Jing Wang for improving the user interface. We also thank Dazhou Guo for many insightful discussions.

References

[1] B. Markovsky and M. Webster, Jr. (In press.) “Theory construction,” in George S. Ritzer (ed.), The Blackwell Encyclopedia of Sociology, 2nd Edition. Malden, MA: Blackwell.

[2] B. Markovsky, “Modularizing Small Group Theories in Sociology,” Small Group Research, vol. 41, no. 6, pp. 664-687, Dec. 2010

[3] K. Aarts, “Parsimonious Methodology,” Methodological Innovations Online Vol. 2, no. 1, pp. 2-10, 2007

[4] H.G. Gauch, “Scientific Method in Practice,” Cambridge Univ. Press, 2003.

[5] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, “Evaluating collaborative filtering recommender systems,” ACM Trans. Inf. Syst., vol. 22, no. 1, pp. 553, 2004

[6] F. Zhang, T. Gong, V. E. Lee, G. Zhao, C. Rong, and G. Qu, “Fast algorithms to evaluate collaborative filtering recommender systems,” Knowledge-Based Systems, vol. 96, pp. 96-103, 2016.

[7] R. J. Mooney and L. Roy, “Content-based book recommending using learning for text categorization,” in Proc. ACM Conf. Digit. Lib., 2000, pp. 195204.

[8] M.J. Pazzani and D. Billsus, “Content-Based Recommendation Systems,” The Adaptive Web, P. Brusilovsky, A. Kobsa, and W. Nejdl, eds., vol. 4321, pp. 325-341, Springer-Verlag, 2007.

[9] M. de Gemmis, P. Lops, C. Musto, F. Narducci, and G. Semeraro, “Semantics-Aware Content-Based Recommender Systems,” in F. Ricci, L. Rokach, and B. Shapira, editors, Recommender Systems Handbook, pp. 119159. Springer, 2015.

[10] R. Burke, “Hybrid Web Recommender Systems,” The Adaptive Web, P. Brusilovsky, A. Kobsa, and W. Nejdl, eds., vol. 4321, ch. 12, pp. 377-408, Springer, 2007.

[11] D. Wang, Y. Liang, D. Xu, X. Feng, and R. Guan, “A content-based recommender system for computer science publications,” Knowledge-Based Systems, vol. 157, pp. 1-9, 2018.

[12] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, “Application of dimensionality reduction in recommender system-a case study,” DTIC Document2000.

[13] D. Cer, M. Diab, E. Agirre, I. Lopez-Gazpio, and L. Specia, “SemEval2017 Task 1: Semantic textual similarity multilingual and crosslingual focused evaluation,” in Proc. 11th Int. Workshop Semantic Eval., Aug. 2017, pp. 114.

[14] E. Agirre, D. Cer, M. Diab, and A. Gonzalez-Agirre, “Semeval-2012 task 6: A pilot on semantic textual similarity,” in Proc. 6th Int. Workshop Semantic Eval. (SemEval 2012), Montreal, Canada, 78, 2012, pp. 385393 [Online]. Available: http://www.aclweb.org/anthology/S12-1051

[15] E. Agirre, D. Cer, M. Diab, A. Gonzalez-Agirre, and W. Guo, “Sem2013 shared task: Semantic textual similarity,” in Proc. 2nd Joint Conf. Lexical and Comput. Semantics (*SEM), 1: Proc. Main Conf. and Shared Task: Semantic Textual Similarity: Assoc. for Comput. Linguist., Atlanta, GA, USA, Jun. 2013, pp. 3243 [Online]. Available: http://www.aclweb.org/anthology/S13-1004

[16] E. Agirre, C. Banea, C. Cardie, D. Cer, M. Diab, A. Gonzalez-Agirre, W. Guo, R. Mihalcea, G. Rigau, and J. Wiebe, “SemEval-2014 Task 10: Multilingual semantic textual similarity,” in Proc. 8th Int. Workshop Semantic Eval. (SemEval-14), Dublin, Ireland, Aug. 2014.

[17] E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Pasca, and A. Soroa, “A study on similarity and relatedness using distributional and wordnet-based approaches,” in Proc. Human Lang. Technol.: Annu. Conf. North Amer. Chap. Assoc. for Comput. Linguist., Boulder, CO, USA, Jun. 2009, pp. 1927

[18] R. Mihalcea, C. Corley, and C. Strapparava, “Corpus-based and knowledge-based measures of text semantic similarity,” in Proc. 21st Nat. Conf. Artif. Intell.. Palo Alto, CA, USA: AAAI Press, 2006, pp. 775780 [Online]. Available: http://dl.acm.org/citation.cfm?id=1597538.1597662

[19] A. Budanitsky and G. Hirst, “Evaluating wordnet-based measures of lexical semantic relatedness,” Comput. Linguist., vol. 32, no. 1, pp. 1347, Mar. 2006.

[20] D. Cer, Y. Yang, S. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, Y. Sung, B. Strope, and R. Kurzweil, “Universal sentence encoder,” CoRR, vol. abs/1803.11175, 2018.

[21] Y. Mao, K. Van Auken, D. Li, C. N. Arighi, P. McQuilton, G. T. Hayman, S. Tweedie, M. L. Schaeffer, S. J. F. Laulederkind, S.-J. Wang, J. Gobeill, P. Ruch, A. T. Luu, J.-j. Kim, J.-H. Chiang, Y.-D. Chen, C.-J. Yang, H. Liu, D. Zhu, Y. Li, H. Yu, E. Emadzadeh, G. Gonzalez, J.-M. Chen, H.-J. Dai, and Z. Lu, “Overview of the gene ontology task at BioCreative IV,” Database, vol. 2014, bau086, 2014.

[22] Y. Wang, N. Afzal, S. Liu, M. Rastegar-Mojarad, L. Wang, F. Shen, S. Fu and H. Liu, “Overview of BioCreative/OHNLP Challenge 2018 Task 2: Clinical Semantic Textual Similarity,” proc. BioCreative/OHNLP Challenge, 2018.

[23] J. Feng and S. Wu, “Detecting near-duplicate documents using sentence level features,” in Proceedings of the International Conference on Database and Expert Systems Applications, 2015.

[24] T. Achakulvisut, DE Acuna, T. Ruangrong, K. Kording, “Science Concierge: A fast content-based recommendation system for scientific publications,” PloS one 11 (7), e0158423, 2016

[25] A. Kadhim, Y. Cheah, I. Hieder, R. Ali, “Improving TF-IDF with Singular Value Decomposition (SVD) for Feature Extraction on Twitter,” in Proc. 3rd. International Engineering Conference. on Developments in Civil & Computer Engineering Applications, 2017

[26] G. N. Lance and W. T. Williams, “Mixed-data classificatory programs I: Agglomerative systems,” Austral. Comput. J., vol. 1, no. 1, pp. 1520, 1967.

[27] J. Davey and E. Burd, “Evaluating the suitability of data clustering for software remodularization,” Proc. Working Conf. Reverse Eng., pp. 268276, Nov. 2000.

[28] S.-S. Chot, S.-H. Cha, and C. C. Tappert, “A survey of Binary similarity and distance measures,” Journal of Systemics, Cybernetics and Informatics, vol. 8, no. 1, pp. 43 48, 2010.

[29] V.I. Levenshtein, “Binary Codes Capable of Correcting Deletions, Insertions and Reversals,” Doklady Akademii Nauk SSSR, vol. 163, no. 4, pp. 845-848, 1965, original in Russiantranslation in Soviet Physics Doklady, vol. 10, no. 8, pp. 707-710, 1966.

[30] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in Proc. Advances Neural Information Processing Systems, 2013, pp. 31113119.

[31] M. E. Wall, A. Rechtsteiner, and L. M. Rocha, “Singular value decomposition and principal component analysis,” in A Practical Approach to Microarray Data Analysis. Norwell, MA: Kluwer, 2003, pp. 91109.

[32] S. Bird, E. Klein, and E. Loper, Natural Language Processing With Python,” 1st ed. Sebastopol, CA: OReilly Media, 2009.

[33] H. Breu, J. Gil, D. Kirkpatrick, and M. Werman, “Linear time Euclidean Distance Transform Algorithms,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 17, pp. 529-533, 1995.

[34] J. R. Bray and J. T. Curtis, “An ordination of the upland forest communities of southern Wisconsin,” Ecological Monographs, vol. 27, no. 4, pp. 325349, 1957. [Online]. Available: http://dx.doi.org/10.2307/1942268

[35] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in Proc. SIGKDD, 2016, pp. 785794.

[36] R. Kohavi, “A study of cross-validation and bootstrap for accuracy estimation and model selection,” in Proc. IJCAI, 1995, pp. 11371145.

[37] I. H. Witten and E. Frank, “Data Mining: Practical Machine Learning Tools and Techniques, 2nd ed. San Francisco, CA: Morgan Kaufmann, 2005.

[38] J. Zobel and A. Moffat, “Exploring the Similarity Space,” ACM SIGIR Forum, vol. 32, pp. 18-34, 1998.

[39] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. “A large annotated corpus for learning natural language inference,” arXiv preprint arXiv:1508.05326, 2015.

[40] H. Jafarkarimi, A. T. H. Sim, and R. Saadatdoost, “A naive recommendation model for large databases,” Int. J. Inf. Educ. Technol., vol. 2, no. 2, pp. 216219, Jun. 2012.

[41] K. Mardia, “Measures of multivariate skewness and kurtosis with applications,” Biometrika, vol. 36, pp. 519530, 1970.