Mining Social Science Publications for Survey Variables

Andrea Zielinski and **Peter Mutschke**
GESIS - Leibniz Institute for the Social Sciences
Unter Sachsenhausen 6-8
50667 Cologne, Germany
[andrea.zielinski,peter.mutschke]@gesis.org

Abstract

Research in Social Science is usually based on survey data where individual research questions relate to observable concepts (variables). However, due to a lack of standards for data citations a reliable identification of the variables used is often difficult. In this paper, we present a work-in-progress study that seeks to provide a solution to the variable detection task based on supervised machine learning algorithms, using a linguistic analysis pipeline to extract a rich feature set, including terminological concepts and similarity metric scores. Further, we present preliminary results on a small dataset that has been specifically designed for this task, yielding a significant increase in performance over the random baseline.

1 Introduction

In face of the growing number of scientific publications, Text Mining (TM) becomes increasingly important to make hidden knowledge explicit. A particular challenge in this regard is to identify research data citations in scholarly publications, due to their wide variety, ranging from quotations to free paraphrases. The problem of detecting dataset references in Social Science publications has been addressed so far by Boland et al. (2012) who mine patterns for discovering dataset citations in full texts to link them to the corresponding entries in a Social Science dataset repository. The recognition, however, has been done just on study name level, in the Social Sciences typically a survey study, e.g. the International Social Survey Programme ISSP. Survey studies, however, usually consist of several hundreds of concepts, so-called variables, each of them representing a single survey question (e.g. *Do you believe in Heaven?*). Therefore, from the perspective of the Social Sciences, having a linkage just to the entire study would not be sufficient to clearly identify the data actually used. For this, identifying the precise variable, the precise subset of variables respectively that was referenced, is strongly needed.

A fine-grained linking between publications and data on the level of variables would have a number of benefits to researchers: It would enable indexing publications by survey variables and discovering publications that discuss the concept of interest (a particular variable). Moreover, it would facilitate a monitoring of the relevance of topical issues (by tracking the use of variables for research) as well as detecting research gaps (by tracking the variables not being addressed by researchers).

The problem, however, is that even though variables are usually assigned a code and a label (e.g. *V39: Belief in life after death* or *V40: Believe in Heaven* from the ISSP 1998 study) as well as the question text from the questionnaire, in practice, authors often do not adhere to citation standards, neither for study names nor for variables. Instead, authors tend to use variations of label and/or question text or combine several variables in one phrase (such as *"...belief in afterlife and Heaven..."* (Neporov and Nepor, 2009)).

In this paper, we introduce the novel task of identifying variables which we define as a multi-label classification task, drawing on ideas from Paraphrase Identification, Citation Matching, and Answer Retrieval in a Question Answering (QA) scenario. Given a set of survey variables, the system needs to examine if one or more of them are mentioned in a text. The task is particularly challenging for the following reasons: The scholarly publications are heterogeneous, covering various styles and topics, and noisy due to pdf-to-text conversion. Moreover, training data is sparse. There-

Proceedings of the Second Workshop on Natural Language Processing and Computational Social Science, pages 47–52,
Vancouver, Canada, August 3, 2017. ©2017 Association for Computational Linguistics

fore, it is crucial to investigate how existing methods in the field of NLP can be applied to our use case. We present a work-in-progress study that seeks to provide a solution to the variable detection task based on supervised ML, using a linguistic analysis pipeline to extract indicative features, ranging from surface-oriented to lexical semantic features.

The overall task can be interpreted either as an information retrieval task, trying to return the most relevant spans of text, as exemplified in TREC QA track (Voorhees, 1999), or as the task to assess the semantic similarity between two (generally very short) text pairs (Agirre et al., 2013). Both approaches can also be combined, i.e. by filtering out good candidates from (possibly huge) document collections in the first stage, and using higher-level semantic processing tools in the second step in order to increase precision.

The paper is organized as follows: Section 2 presents related work, Section 3 describes the Social Science use case, Section 4 reports on two basic approaches to the task, summarizing their underlying resources and tools, Section 5 shows the experiments and discusses the results. Finally, Section 6 draws the conclusions and shows future directions.

2 Related Work

Variable Detection is a new task, yet closely related to several existing lines of work in the field of NLP. At its core it is detecting the similarity between sentences, involving the complex task of textual entailment recognition and paraphrase detection at the upper end of the spectrum and string matching, prominent for, *e.g.*, detecting plagiarism, at the lower end of it.

In the Pascal Challenge *Recognizing Textual Entailment (RTE)* (Dagan et al., 2006), QA systems have been designed to identify texts that entail a hypothesized answer (T) to a given question (H). The best results were obtained by lexically-based systems without deeper semantic reasoning, relying on ML techniques, similarity measures (string, lexical and syntactic-based), knowledge resources (e.g., WordNet, paraphrase corpora) and linguistic analysis (*e.g.* Punyakanok et al. (2004) compute the tree edit distance between the dependency trees of the question and answer, and Bouma et al. (2005) use deep syntactic parsing and distributional similarities from external corpora). Even though results to the RTE task in general were modest with accuracy scores between 50-60%, for specific task settings, they could bring accuracy gains: Harabagiu (2006) report an increase in performance from 30.6% to 42.7% on an open-domain QA task.

An important component for any QA system is sentence retrieval, since answers occur locally in a text. The systems' performance is generally evaluated by means of the mean reciprocal rank (MRR) of top k sentences retrieved as answers to a question. The problem of re-ranking pairs of short texts has been addressed by Severyn et al. (2015) who build a convolutional neural network architecture. When augmenting the deep learning model with word overlap features the model achieves an improvement of 3% in MAP and MRR on the TREC QA task. For the same task, an increase in performance could also be observed by Bordes et al. (2014) by adopting deep learning techniques. The authors set up a compositional embedding model, projecting question and answer pairs into a joint space. Kusner et al. (2015) define a distance metric between text documents, *i.e.* the Word Mover's Distance. The metric utilizes *word2vec* word embeddings pre-trained on the Google News Corpus to address the vocabulary mismatch problem. The authors report that WMD achieves an error reduction of up to 10% for the k-nearest neighbor document classification task as compared to traditional approaches, outperforming LDA.

An overview of the plagiarism detection competition in PAN-PC11 is given in Potthast (2011). Best results on extrinsic plagiarism, with a focus on cases made up of $<$ 50 words, achieve 14% recall and 70% precision (evaluated on a character basis). A more fine grained typology of plagiarism is given in (cf. Baron (2013)) who reports that while *copy&paste* plagiarism can be detected reliably using VSMs, fingerprinting or substring matching methods, cases involving the recognition of text segments that are paraphrases, are extremely hard to detect. On the P4P corpus - a subset of the PAN-PC-10 Corpus - a modest recall of 12% could be achieved by Costa-Jussà (2010) for the best performing system.

3 Task Description

Identifying mentions of survey variables in texts can be defined as a multi-label classification problem: given a set of sentences S $\subseteq \{s_1, .., s_i\}$ and variables V $\subseteq \{v_1, .., v_j\}$, we need to build a classifier function $h : S \rightarrow V$. Each variable v has a unique label (*i.e.* a class) characterizing its semantics. Each sentence s is represented by a single instance which can be associated with one (or more) class label(s), including *non-related* as a label. Usually, the number of labels assigned to s is relatively small. Since the link between a publication and a study has been established beforehand, the set of labels can be reduced to those that occur in the respective study.

A gold standard corpus entitled *ALLBUS-English* and *ALLBUS-German* has been compiled and annotated by two Social Sciences students. In doing so, they have taken the specific document context as well as dependencies among variables belonging to the same study into account. Identical survey variables (ca. 8%) have been clustered beforehand. The corpus is composed of sentences labeled with any of the 62 (88) variables from the underlying survey studies, yielding 66 (98) sentences classified as relevant, while the vast majority of sentences is unrelated, i.e. 4.585 (8.351) sentences for English and German respectively. Average density of labels is 1.02 and average length of a variable text is about 14.3 tokens per sentence. A typical example showing how variable references can differ from their data catalog entry is provided below:

> **Reference**: "*Foreigners should not be allowed to engage in political activities*."
> **Survey_Variable$_{\text{v45-ALLBUS-ZA4500}}$**:
> "Please tell me for each statement to what extent you agree with it. [..]. *Foreigners* living in Germany *should be prohibited from taking part in any kind of political activity* in Germany."

A first empirical investigation revealed different types of variable references, most prominently:

- Citations, reported speech, *i.e.*, either exact copies of a text fragment or marked by quotation marks (such as "Foreigners" from the above example)
- Lexical modifications, due to synonym substitution or compounding, along with negation: "should be prohibited" (Survey) vs. "should not be allowed" (Reference), "taking part in" (Survey) vs. "to engage in" (Reference)
- Morphological variations: "political activity" (Survey) vs. "political activities" (Reference)
- Trend to shorten and summarize the variable: "belief in life after death" (Survey) vs. "belief in afterlife" (Reference)
- Word order modifications along with verb/noun conversions and omissions: "life after" (Survey) vs. "afterlife" (Reference), omission of "in Germany" in the above example.

4 Approaches for Variable Detection

In our experiments, we tested (**A**) a supervised ML model based on a Bag of Words (BoW) representation, using linguistic and conceptual features, and integrating external knowledge resources, and (**B**) a supervised ML model using real-valued feature vectors derived from computing semantic similarity metrics for pairs of variables and sentences. In both approaches, **A** and **B**, documents are first preprocessed and the variable lists are retrieved from the data catalog. Then, a rich set of features is computed from sentences and variables.

4.1 Feature Extraction

For pre-processing, we use a pipeline of tools from DKPro (de Castilho and Gurevych, 2014) that supports tokenization, lemmatization, part-of-speech tagging and Named Entity Recognition. For text segmentation, *i.e.* extracting sentences from sections and paragraphs, we use a pdf-to-text converter. Titles as well as tables are largely ignored.

For approach **A** we integrate general lexical resources as well as the thesaurus for the Social Sciences *TheSoz* (Zapilko et al., 2013), extracting the following features from sentences and variables:

- Tokens, lemmas, PoS using Schmid (1995)
- Named Entities using *Stanford NER* (Finkel et al., 2005; Faruqui and Padó, 2010)
- Term filter, selecting lemmas with PoS=Noun, Verb, Adjective (idf-weighted)
- Keyword terms, synonyms and hypernyms from *TheSoz*
- Synonyms, hypernyms as well as derivational variants from *WordNet* (Fellbaum, 1998; Hamp and Feldweg, 1997)

For **B** we rely on a set of similarity distance metrics provided by DKPro Similarity (Bär et al., 2013) and by the Evaluation Framework for Statistical Machine Translation. In particular, the *METEOR* metric has proven to yield competitive results in the paraphrase detection task (Pado et al., 2014). Extracted features from all the S-V-pairs are:

- *DKPro Similarity* metrics such as character and word n-grams (1,2,3,4), greedy string tiling, longest common subsequence (Bär et al., 2013).
- *BLEU*: maximum n-gram order of 4 (Papineni et al., 2002).
- *METEOR*, using the standard setting with normalization and all variants *exact, stem, synonym and paraphrase* (Banerjee and Lavie, 2005) with extended *DBnary* for German (Elloumi et al., 2015).

4.2 Classification Algorithms

For approach **A**, we use a BoW representation of features from 4.1 and experiment with 3 learning algorithms from the ML framework WEKA (Witten et al., 1999), Naive Bayes, KNN and SVM linear. In order to rank candidate sentences, *i.e.* all sentences not classified as *non-related*, we use the Nearest Neighbor algorithm which returns the closest instances for V based on majority voting. KNN already provides a simple, yet effective solution to the multi-label problem.

In **B**, similarity is encoded in the similarity scores (cf. 4.2). Generally, for a new task, finding the best measures and thresholds is difficult, since no prior heuristics exist. In order to find out which scores correlate most with human judgments, we computed the Pearson correlation coefficient $r_{S,V}$.

5 Experiments and Results

5.1 Supervised ML model based on BoW (A)

The variables' texts were used to train a set of classifiers, resulting in one classifier per variable. For our experiments, we first tested one single feature set at a time, in order to determine which feature sets are most effective for the task. Then, we also combined all features to find out if this increases classifier performance, iterating over the set of ML algorithms. In order to be able to detect irrelevant sentences, we introduced some noise (1% *non-related*) from withheld sentences. Testing was carried out on the entire German and English ALLBUS corpus (disjoint from the training set).

Results are given in Table 1, showing a significant increase in recall (by a factor of 14) and precision (by a factor of 6) for English. Likewise for German, recall could be enhanced (by a factor of 9) and precision (by a factor of 5) over the random baseline. Results obtained for English are consistently above the keyword match baseline (cf. (Light et al., 2001)).

An interesting finding is that domain-specific *TheSoz* terms achieve a relatively high performance, in particular for German. In combination with *WordNet* terms, synonyms bring most gain, followed by hypernyms and derivations. Also, the performance of classifiers varies considerably. We observed that when running multiple classifiers in an ensemble, different result sets could be retrieved, increasing recall. Adding features derived from the answers of the variables improved recall slightly. Overall, the percentage of missed items is relatively high, because key correspondences were not always detected. For instance, the system failed to bridge from *people from EU countries coming to work here* to *EU workers* in the example below.

> **Reference**: "To measure anti-immigrant sentiments, [..] regarding citizens' beliefs about immigration for four groups: asylm seekers, *EU workers*, non-EU workers and ethnic Germans. []"
> **Survey_Variable_v121-ALLBUS-ZA3450**: "[]. What is your opinion about this for *people* from *EU countries* coming *to work* here?"

Furthermore, we applied NN search and ranking algorithm on the combined feature set up to rank 100. Results reveal that most mentions of variables are among the top 10. Overall, MAP is higher for English than for German due to the higher coverage of lexico-semantic resources. Note that the class distributions also vary.

5.2 Supervised ML model on similarity metrics (B)

For this experiment, we aimed for a balanced dataset consisting of all positive pairings (from our gold standard) and adding randomly generated combinations of S-V pairings to constitute the *non-related* class (with 10-fold cross-validation).

Corpus	**ALLBUS English**						**ALLBUS German**					
Classifier	KNN		Naive Bayes		Linear SVM		KNN		Naive Bayes		Linear SVM	
Performance	**MAP**	**MAR**	**MAP**	**MAR**	**MAP**	**MAR**	**MAP**	**MAR**	**MAP**	**MAR**	**MAP**	**MAR**
Token	0.03	0.08	0.03	0.03	0.04	0.05	0.03	0.06	0.01	0.01	0.02	0.04
Lemma	0.06	0.06	0.03	0.03	0.06	0.06	0.02	0.06	0.03	0.03	0.02	0.03
Terms	0.06	0.09	0.02	0.03	0.05	0.06	0.03	0.08	0.01	0.01	0.03	0.09
NER	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.03	0.01	0.01	0.02	0.03
TS-S	0.04	0.08	0.02	0.02	0.04	0.08	**0.05**	**0.10**	0.01	0.01	**0.05**	**0.10**
WN-S	0.08	0.11	0.03	0.10	0.07	0.12	0.04	0.07	0.02	0.06	0.04	0.07
WN-H	0.06	0.13	0.02	0.05	0.06	0.13	0.03	0.07	0.02	0.03	0.02	0.08
WN-D	0.06	0.14	0.02	0.05	0.07	0.13	0.03	0.04	0.01	0.01	0.03	0.08
ALL	0.07	0.14	**0.09**	**0.22**	**0.09**	0.15	**0.05**	0.07	0.01	0.04	0.04	0.07

Table 1: Performance on ALLBUS for different Feature Sets (Terms; NER: *Stanford NER*; TS-S: *TheSoz*; WN-S: *WordNet Synonyms*; WN-H: *WordNet Hypernyms*; WN-D: *WordNet Derivations*; *All Features combined*; measures are: *Macro Average Precision (MAP); Macro Average Recall (MAR); Random Baseline English 0.016; Random Baseline German 0.011)*

Then, for all German and English pairs, the individual similarity scores for different standard metrics were computed and fed into a linear regression classifier.

Results are listed in Table 2 and indicate that overall Pearson correlation scores are relatively low - in particular for German (betw. 0.06 and 0.62). Surprisingly, robust metrics like Levenshtein yield a relatively high correlation score, outranking *METEOR*. Due to its ability to detect citations and deal with noisy input, results are overall better, while term expansion/weighting and unigram alignment cannot compensate for this.

Metrics	**E** $r_{S,V}$	**G** $r_{S,V}$
$LSSC$	0.92678	0.6216
LC	0.78116	0.5986
$JWSSC$	0.7332	0.5421
GTS_3	0.42132	0.4039
$JSSC$	0.22879	0.3586
GTS_2	0.28602	0.3379
$LCSC$	0.52536	0.3361
$BLEU$	0.20972	0.2648
MET_{ssp}	0.75103	0.2413
$ngram_2$	0.03662	0.2315
$ngram_3$	0.74195	0.1862
M_{ess}	0.40991	0.1666
$ngram_4$	0.09381	0.1478
GTS_4	0.75164	0.0662

Table 2: Pearson Correlation Scores (G: German; E: English; $LSSC$: Levenshtein Second String Comparator; LC: Levenshtein Comparator; $JWSSC$: JaroWinkler SecondString Comparator; GTS_*: Greedy String Tiling; $JSSC$: Jaro Second String Comparator; $BLEU$; MET_{ssp}: Meteor stem-synonym-paraphrase; $LCSC$: Longest Common Subsequence Comparator; $n - gram*$; MET_{ess}: Meteor exact-stem-synonym).

6 Conclusion and Future Work

On the variable detection task, our first experiments give insights into the performance for various NLP methods. The choice of features was motivated by empirical corpus investigations. While the dataset is relevant for the task, it is still too small to train and develop robust ML classifiers. Yet, evaluating the two approaches with different parameter settings and testing them individually provides interesting results on their own which we will use for future work. First, we will elaborate on the BoW approach, by a) integrating novel language modeling techniques (such as word embedding) to increase recall and b) enhancing term weights from external resources, since terminology proved to be important for retrieving variables. Second, we will devise specialized classifiers for the recognition of citations and reported speech for which string similarity based classifiers are well suited. Last but not least, we will adapt *METEOR* to better fit the task, e.g. optimizing the penalty score and matching, because it has a high potential for disambiguating related variables.

Acknowledgments

This work was supported by the European Union's Horizon 2020 research and innovation programme (H2020- EINFRA-2014-2) under grant agreement No. 654021 (OpenMinTeD[1]).

References

E. Agirre, D. Cer, M. Diab, A. Gonzalez-Agirre, and W. Guo. 2013. Sem 2013 shared task: Semantic textual similarity, including a pilot on typed-similarity. In **SEM 2013: The Second Joint Conference on Lexical and Computational Semantics*.

S. Banerjee and A. Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation

[1] http://openminted.eu/

with human judgments. In *Proc. of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*. volume 29, pages 65–72.

D. Bär, T. Zesch, and I. Gurevych. 2013. Dkpro similarity: An open source framework for text similarity. In *ACL (Conference System Demonstrations)*. pages 121–126.

A. Barrón-Cedeño, M. Vila, M. Martí, and P. Rosso. 2013. Plagiarism meets paraphrasing: Insights for the next generation in automatic plagiarism detection. *Computational Linguistics* 39(4):917–947.

K. Boland, D. Ritze, K. Eckert, and B. Mathiak. 2012. Identifying references to datasets in publications. In *International Conference on Theory and Practice of Digital Libraries*. Springer, pages 150–161.

A. Bordes, S. Chopra, and J. Weston. 2014. Question answering with subgraph embeddings. *ArXiv preprint arXiv:1406.3676* .

G. Bouma, J. Mur, G. Van Noord, L. Van Der Plas, and J. Tiedemann. 2005. Question answering for dutch using dependency relations. In *Workshop of the Cross-Language Evaluation Forum for European Languages*. Springer, pages 370–379.

M. Costa-Jussà, R. Banchs, J. Grivolla, and J. Codina. 2010. Plagiarism detection using information retrieval and similarity measures based on image processing techniques-lab report for pan at clef 2010. In *CLEF (Notebook Papers/LABs/Workshops)*.

I. Dagan, O. Glickman, and B. Magnini. 2006. The pascal recognising textual entailment challenge. In *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*, Springer, pages 177–190.

R. de Castilho and I. Gurevych. 2014. A broad-coverage collection of portable nlp components for building shareable analysis pipelines. In *Proc. of the Workshop on Open Infrastructures and Analysis Frameworks for HLT (OIAF4HLT) at COLING*. pages 1–11.

Z. Elloumi, H. Blanchon, G. Serasset, and L. Besacier. 2015. Meteor for multiple target languages using dbnary. In *MT Summit 2015*.

M. Faruqui and S. Padó. 2010. Training and evaluating a german named entity recognizer with semantic generalization. In *KONVENS*. pages 129–133.

Ch. Fellbaum. 1998. *WordNet*. Wiley Online Library.

J. Finkel, T. Grenager, and Ch. Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proc. of the 43rd annual meeting on association for computational linguistics*. pages 363–370.

B. Hamp and H. Feldweg. 1997. Germanet. a lexical-semantic net for german. In *Proc. of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*. pages 9–15.

S. Harabagiu and A. Hickl. 2006. Methods for using textual entailment in open-domain question answering. In *Proc. of the 21st International Conference on Computational Linguistics*. pages 905–912.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *International Conference on Machine Learning*. pages 957–966.

M. Light, G. Mann, E. Riloff, and E. Breck. 2001. Analyses for elucidating current question answering technology. *Natural Language Engineering* 7(04):325–342.

O. Neporov and Z. Nepor. 2009. Religion: An unsolved problem for the modern czech nation. *Czech Sociological Review* 45(6):1215–1237.

S. Pado, A Stern, B. Magnini, R. Zanoli, and I. Dagan. 2014. Excitement open platform: Architecture and interfaces. In *ACL (System Demonstrations)*. pages 43–48.

K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proc. of the 40th annual meeting on association for computational linguistics*. pages 311–318.

M. Potthast, A. Eiselt, L. A. Barrón Cedeño, B. Stein, and P. Rosso. 2011. Overview of the 3rd international competition on plagiarism detection. In *CEUR Workshop Proceedings*. CEUR Workshop Proceedings, volume 1177.

V. Punyakanok, D. Roth, and W. Yih. 2004. Mapping dependencies trees: An application to question answering. In *Proceedings of AI&Math 2004*. pages 1–10.

H. Schmid. 1995. Treetagger - a language independent part-of-speech tagger. *Proc. of Int. Conference on New Methods in Language Processing* 43:28.

A. Severyn and A. Moschitti. 2015. Learning to rank short text pairs with convolutional deep neural networks. In *Proc. of the 38th International ACM SIGIR Conf. on Research and Development in Information Retrieval*. ACM, pages 373–382.

E. Voorhees. 1999. The trec-8 question answering track report. In *Trec*. volume 99, pages 77–82.

I. Witten, E. Frank, L. Trigg, A. Hall, G. Holmes, and S. Cunningham. 1999. Weka: Practical machine learning tools and techniques with java implementations. .

B. Zapilko, J. Schaible, P. Mayr, and B. Mathiak. 2013. Thesoz: A skos representation of the thesaurus for the social sciences. *Semantic Web* 4(3):257–263.