American Journal of Information Science and Technology
2022; 6(1): 1-7
http://www.sciencepublishinggroup.com/j/ajist
doi: 10.11648/j.ajist.20220601.11
ISSN: 2640-057X (Print); ISSN: 2640-0588 (Online)

Training Data Improvement by Automatic Generation of Semantic Networks for Bias Mitigation

Roman Englert[1, 2], Jörg Muschiol[1]

[1]Computer Science, FOM University of Applied Sciences, Essen, Germany

[2]New Media and Information Systems, Faculty III, Siegen University, Siegen, Germany

Email address:
roman.englert@fom-net.de (R. Englert)

To cite this article:
Roman Englert, Jörg Muschiol. Training Data Improvement by Automatic Generation of Semantic Networks for Bias Mitigation. *American Journal of Information Science and Technology.* Vol. 6, No. 1, 2022, pp. 1-7. doi: 10.11648/j.ajist.20220601.11

Received: February 20, 2022; **Accepted**: March 16, 2022; **Published**: March 29, 2022

Abstract: The significance of Bias Detection has increased appreciably, due to the increased application of AI. Although syntactic bias is well explored with statistical techniques, there remains semantic bias challenge like for example, Google's face recognition which excludes colored people. Human expertise is required to detect semantic bias, e.g., for the application of the root-out-bias method. We propose a further automatization to this laborious method, based on the Training Data Improvement for Bias Mitigation (TDIBM). The concept, is to automatically construct a Semantic Network (SN) from the domain description of the training. For the semantic network nouns are extracted. As a second step, synonyms and semantically similar nouns are searched, e.g. in dictionaries, and added to the SNs. As a result, the SN contains nouns that enhances the given domain, with previously unknown knowledge. This SN can be used to check with, e.g., the root-out bias method, whether the training sample is biased, or not. Should the training sample be biased, then the corresponding nouns from the SN can be added to the training sample set to mitigate the bias. The newly developed method, TDIBM is evaluated twofold: Firstly, with the description of the COMPAS system, which is a case management and decision support tool used by U.S. courts to assess the likelihood of a defendant becoming a recidivist. Secondly, an autonomous driving domain is applied, to investigate accidental driving of a Tesla car. Here TDIBM detected among many new features, including one to solve ambiguous scene interpretations for autonomous driving vehicles.

Keywords: Semantic Bias Detection, Bias Mitigation, Semantic Networks, Semantic Similar Words, AI, Bias, Bias Detection, Training Sample

1. Introduction

AI becomes more and more an integral part of Software engineering and applications. For most AI systems, a huge amount of data is required for training. Unfortunately, training samples often have a bias, or may be retrained on purpose with bias during their operations [13]: An example of the latter is the chat bot Tay from Microsoft that has been trained by users with gender-bias and racism terms [14]. The challenge is to detect this kind of semantic bias in order to avoid misuse of AI systems and improper training samples, e.g. the face recognition training sample from Google that is lacking in skin color [13]. The goal is to further automate the root-out bias method that requires significant human intervention, for semantic bias detection, in order to make it applicable to AI training samples. The human expert has to choose, and to add features to the domain in order to mitigate bias. The success of this, depends on the experience and the proper intuition, of the user, to find the missing features / training samples. A more systematic approach is based on the concept that the nouns of a domain description enable one to determine semantically similar knowledge that mitigates bias. As an example, consider Google's face recognition Software which recognizes faces, but only if they are white-colored [28]. This bias could be overcome by adding the feature 'color/tone' that is semantically similar to 'skin' and is found by querying Oxford's dictionary for similar words of 'skin' [29].

During the last decades various approaches were developed for the representation, and acquisition of semantic information: During the 80th and 90th systems for logic

inferences based on predicate logic were investigated [1, 2]. These systems require a manual design of domains, and thus they are restricted in the complexity that could be modeled. Another promising approach was the use of semantic networks for the representation of semantic knowledge [3, 4]. However, they also have to be modeled by experts, in order to achieve proper domains. In this paper a new approach TDIBM: to build automatically semantic networks (SN) is proposed and investigated.

The idea is to extract words (nouns) from a text, and to group them based on text frames, indicating that the extracted words belong semantically together. An example for a text frame is a paragraph in a document. Alternative approaches for word distances using AltaVista or the Word Mover Distance (WMD) are provided [9, 11]. The grouping of the extracted words is done by building a semantic network. Then, in a second step, synonyms are added to the semantic network. The constructed SN contains so far only knowledge (word synonyms) from the original text paragraph. This limitation can be overcome, by adding new knowledge to the SN, based on the synonyms: Semantic similar words are searched in the Web using tools like WordNet [12] or Semenov's and Arefin's English word frequency list [31]. The method of applying a Webcrawler is described including applications [5-8].

The paper is organized as follows: In Section 2 related work is described. The developed and evaluated approach TDIBM for the construction of semantic models, which are semantic networks (SN) using frame-based text blocks (FTB) consists of two main steps (Section 3): Firstly, the noun collection phase (NCP), where nouns are extracted from a text. Secondly, the similarity collection phase (SCP), for the collection of nouns using a similarity measure. In both steps a semantic network of the selected words is built. Sections 4 and 5 contain two evaluations of TDIBM. The first is based on the biased system COMPAS (Correctional Offender Management Profiling for Alternative Sanctions) which is a case management and decision support tool used by U.S. courts, to assess the likelihood of a defendant becoming a recidivist [15, 16, 24, 25]. And as second evaluation, the Tesla car accident in 2016 [26], where the autonomous driven car did not recognize a white van. This fatal accident is well investigated and shows the limitations of today's AI computer vision and sensor systems [33]. The training data-set for the AI-based recognition system is investigated for features that it should address. The paper concludes with a summary and an outlook for extending the at hand approach TDIBM.

2. Related Work for Word Similarities

The related work is considered for both NCP and SCP. The NCP is the phase where nouns are collected using a grammar tree parser [20]. In the SCP phase, semantically similar nouns are searched. Semantic similarity can be defined based on word distances or frequencies: The earliest and probably best known approaches for determining word distances and frequencies are IF-TDF and Word2Vec [18, 30]. Both approaches need to be recomputed if the underlying word text database is changed, e.g. enhanced by new texts. This means, that all computations done so far are need to be renewed, also. Islam and Inkpen describe a method for word frequency information based on Alta Vista Advanced Search, which provides information about words, and on how many documents contain them [11, 22]. This method can be applied to the at hand frame-based text search: Here each paragraph of a text needs to be considered as a document. Another approach, is the Word Mover Distance (WMD), proposed by Kusner et al. [9]: This is based on a distance function. This distance function between text documents measures the dissimilarity between two text documents, as a distance of how much one word needs to be moved to reach the position in the other document. In order to compute the distance, they apply so-called word vectors, e.g., *vec(Berlin)* is close to *vec(Potsdam)*. A shortcoming is that the word vector approach requires a trained model for the probability of neighboring words. This can be done with some effort by, e.g., a neural network, as proposed by Mikolov et al. [18].

Semenov and Arefin provide an English word frequency list which contains 2,184,780 different words and is generated from 1,947,152,902 words in the English Wikipedia (March 2019) [31]. An alternative approach is based on word distances: Song et al. use WordNet to determine synonyms using semantic networks [12, 19, 32]. In WordNet semantically similar words are nearer together than less semantically similar words. Semantically similar words are synonym sets with pointers to further synonym sets. Islam and Inkpen propose also methods for the collection of similar words (SCP) to the nouns from the NCP [11]: Various corpus-based similarity measures (CBS) that rely on logarithmic measures are provided. CBS is applied in order to define the similarity of two words: Islam and Inkpen provide several CBS measures for which, e.g., AltaVista Advanced Search can be applied [22]. AltaVista provides the numbers of documents that contain both given words, and additionally, the number of documents that contain each word separately. Thus, the CBS of two words a and b is defined as follows:

$$cbs1(a,b) = \frac{hits(a\ AND\ b)}{hits(a)\cdot hits(b)} \quad (1)$$

where *hits* denotes the number of word occurrences. Another source for word frequencies and contexts is the British National Corpus [23], alternatively, frequencies can be derived from the Word Frequency Data in iWeb Corpus [27]. For the latter, the hits of a and b are known, and thus, the following CBS can be applied:

$$cbs2(a,b) = \frac{hits(a)-hits(b)}{hits(a)} \quad (2)$$

assuming that $hits(a) \geq hits(b)$. This similarity measure provides the percentage deviation of the given words and requires a lower threshold that may be different for word pairs. As a corpus, they use the British National Corpus

(BNC) with more than 100 million words [34]. A large corpus overcomes the need to re-compute the distance database in case of adding new texts. Lin [10] furthermore provides an information-theoretic similarity measure that is based on entropy. Entropy-based deviation of words is described and evaluated in recent research [10, 17]. The challenge here is that a statistical model about word frequencies must be known in advance, which is not the general case for specific domains. As an example consider the probability for the word "king" in a domain for the game chess: The probability for the word "king" in a dictionary is known, but not the occurrence in a specific domain like chess.

In the following, we rely on Semenov's and Arefin's English word frequency list for word distances of nouns [31].

3. The Approach TDIBM: Noun and Similarity Collection Phases

Within the NCP nouns are extracted from a FTB using a grammar tree parser, e.g., the Link Grammar Parser from Lafferty et al. [20]. For an illustration we use the first part of an abstract from the paper about the aforementioned chat bot Tay [14]: "*In 2016, Microsoft launched Tay, an experimental artificial intelligence chat bot. Learning from interactions with Twitter users, Tay was shut down after one day because of its obscene and inflammatory tweets. This article uses the case of Tay to re-examine theories of agency. How did users view the personality and actions of an artificial intelligence chat bot when interacting with Tay on Twitter? Using phenomenological research methods and pragmatic approaches to agency, we look at what people said about Tay to study how they imagine and interact with emerging technologies and to show the limitations of our current theories of agency for describing communication in these settings.*" The extracted nouns are intelligence, bot, interactions, users, obscene, tweets, article, case, theories, agency, user, personality, actions, intelligence, bot, research methods, approaches, agency, people, technologies, limitations, theories, agency, communication and settings. The nouns intelligence, bot, agency, users and theories occur more than once and thus duplicates are removed, resulting in 19 different nouns. Nouns may also be compound like "research methods" and will be counted as one (compound) noun. Additionally, for the further processing the singular of extracted nouns like "interaction_s" is used (Part 1).

Part 1. Extracted nouns from the research paper about the chat bot Tay [14] without duplicates and singular.

intelligence, bot, interaction, user, obscene, tweet, article, case, theory, agency, personality, action, research_method, approach, people, technology, limitation, communication, setting.

Part 2 shows the algorithm for the NCP phase, where nouns are extracted from a FTB and then, the genitive and duplicates are removed. Finally, the nouns are inflected to singular. Note, the extraction of nouns requires an individual grammar model for each natural, where the applied grammar tree parser can remain the same. The 19 extracted nouns will be the root for enhancing the constructed semantic networks for each noun (Part 1).

Part 2. Algorithm for the NCP phase.

Algorithm NCP:

1. Define the FTB in a given text, e.g., a paragraph
2. Extract the starting nouns from the FTB using a grammar parser, e.g., the Link Grammar Parser [20]
3. Remove grammar's genitive of the nouns
4. Remove duplicates from the nouns
5. Inflect nouns to singular

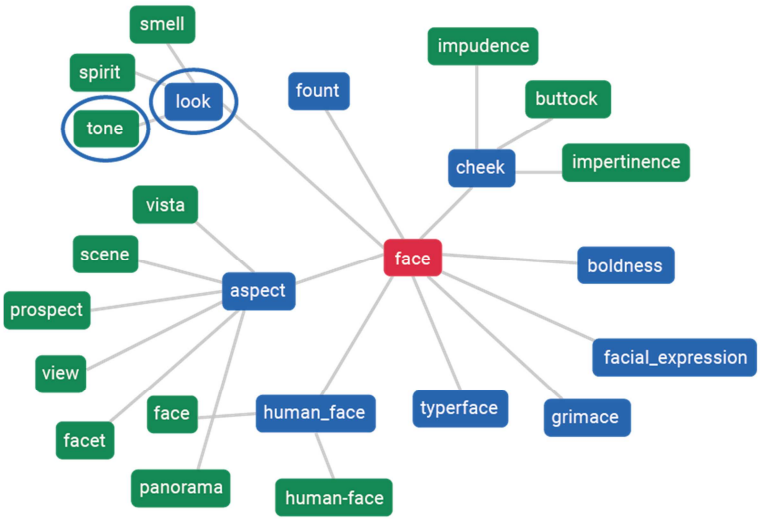

Figure 1. *Expanding a SN with SCP.*

For the selected nouns, semantic similar nouns (synonyms) are collected using, e.g., the Oxford dictionary [21]. These synonyms are called first-order synonyms (FOS) and will be added to a SN. For each outer node of the SN, synonyms can be collected again, the so-called second-order synonyms (SOS) that will also be added to the SN (Figure 1). The

computational complexity of the NCP algorithm (Part 2) is bounded by $O(n^2)$, with n being the maximum number of synonyms.

In the second step, Semenov's and Arefin's word frequency list [31] is applied to select synonyms for the noun *face* (Figure 1): The noun *look* has been added as FOS (blue circle) and the similar noun *tone* as SOS (green circle).

The similarity of two words in a SN can be measured by the minimum path length (MPL) [19]: The MPL varies from zero for FOS and SOS up to infinite when the SN has an infinite number of edges. The similarity of two words *a* and *b* is one for equal words, i.e., MPL is zero, or down to zero, when the path length is infinite. As an example, consider the FOS (Figure 1), where the MPL is zero, since the blue-circled nouns are in the same synonym set. Additionally, the distance between the starting nouns and FOS is one, and between the starting nouns and SOS two. More formally, the similarity is defined as [12]:

$$sim(a,b) = \frac{1}{dis(a,b)+1} \quad (3)$$

where $sim(a, b)$ denotes the semantic similarity of the words a and b, and $dis(a, b)$ is the MPL between them. Then, the similarity decreases as the distance increases. In the above example is *dis(face,look) = 1* and *dis(face,tone) = 2*. Thus, the similarity of *face* and *look* is with *sim(face,look) = 1/2* greater than the similarity of *face* and *tone (sim(face,tone) = 1/3)*. The MPL is applied to Semenov's and Arefin's English word frequency list [31].

Part 3. Algorithm SCP: Expanding the NCP SNs with SCP.

Algorithm SCP:

1. For each noun from NCP do
   1) Search similar nouns from a dictionary with a distance of one in Serenov's and Arefin's word frequency list
   2) Add the similar nouns as FOS to the SN of the starting noun
   3) Search for SOS of the new FOS and add them to the SNs
2. If more similar nouns are required, i.e. more new knowledge is needed, then repeat step 1.

The SCP algorithm for expanding the NCSP SNs is depicted in Part 3. After the first two iterations (NCP and SCP) new knowledge, i.e. new nouns, have been found, namely e.g. the nouns *look* and *tone* (Figure 1). This feature was missing in the training data in order to avoid bias by not recognizing darker skin.

The computational complexity of the SCP algorithm is bound by the size of the SN and the number of nouns to be added, which is expected to be less than the size of the SN, resulting in $O(n^2)$. In the following sections TDIBM is applied to two real cases.

4. Evaluation Using COMPAS

For the evaluation the description of the system COMPAS (Correctional Offender Management Profiling for Alternative Sanctions) is used [24, 25]: COMPAS is a case management and decision support tool used by U.S. courts to assess the likelihood of a defendant becoming a recidivist [25]. Figure 2 contains the three types of defendants: Only the type "Pretrial release" is a candidate for being released from prison, and the other two types "General recidivism" and "Violent recidivism" are not. COMPAS is not a fully transparent system: The „Violent Recidivism Risk Score", the formula to compute the risk is public, but not the used training data for the neural network. The problem with COMPAS is that race is not a variable considered by COMPAS, but reports emerged that COMPAS is racially biased [25], since the risk types are irregularly distributed relative to skin color.

Figure 2. *Three risk types of recidivism [24].*

The following steps are applied to perform the evaluation and improve the feature set of COMPAS' training data:

1. Select FTB for COMPAS from the paper [25] and select nouns
2. Compute NCP (FOS and SOS) and construct SNs
3. Enhance NCP by SCP and also the SNs

As FTB the following text part from the introduction of the paper [25] is applied (step 1): "*The recidivism prediction component of COMPAS—the recidivism risk scale—has been in use since 2000. This software predicts a defendant's risk of committing a misdemeanor or felony within 2 years of assessment from 137 features about an individual and the individual's past criminal record*". As preprocessing plurals and the grammar's genitive are removed from the FTB (see Part 4). Finally, duplicates are treated only once, e.g. "individual".

Part 4. Extracted nouns from the FTB (cursive) without grammar's genitive and without duplicates.

The *recidivism prediction component* of COMPAS—the *recidivism risk* scale—has been in use since 2000. This *software* predicts a *defendant*('s) *risk* of committing a *misdemeanor* or *felony* within 2 *years* of *assessment* from 137 *features* about an *individual* and the (individual's) past *criminal record.*

In step 2, the NCP, similar nouns are collected using Semenov's and Arefin's word frequency list with MPL = 1: For the 18 initial nouns there were 83 FOS synonyms found with TDIBM (Figure 3, blue words) and then in step 3 there were 98 SOS nouns found (Figure 3, green words) using Semenov's and Arefin's word frequency list [31]. The distance for the FOS nouns from the initial noun *recidivism* is one and two for the SOS nouns (Figure 3). The resulting SN contains in total 83 + 98 = 181 new nouns, e.g. *race*, representing new knowledge.

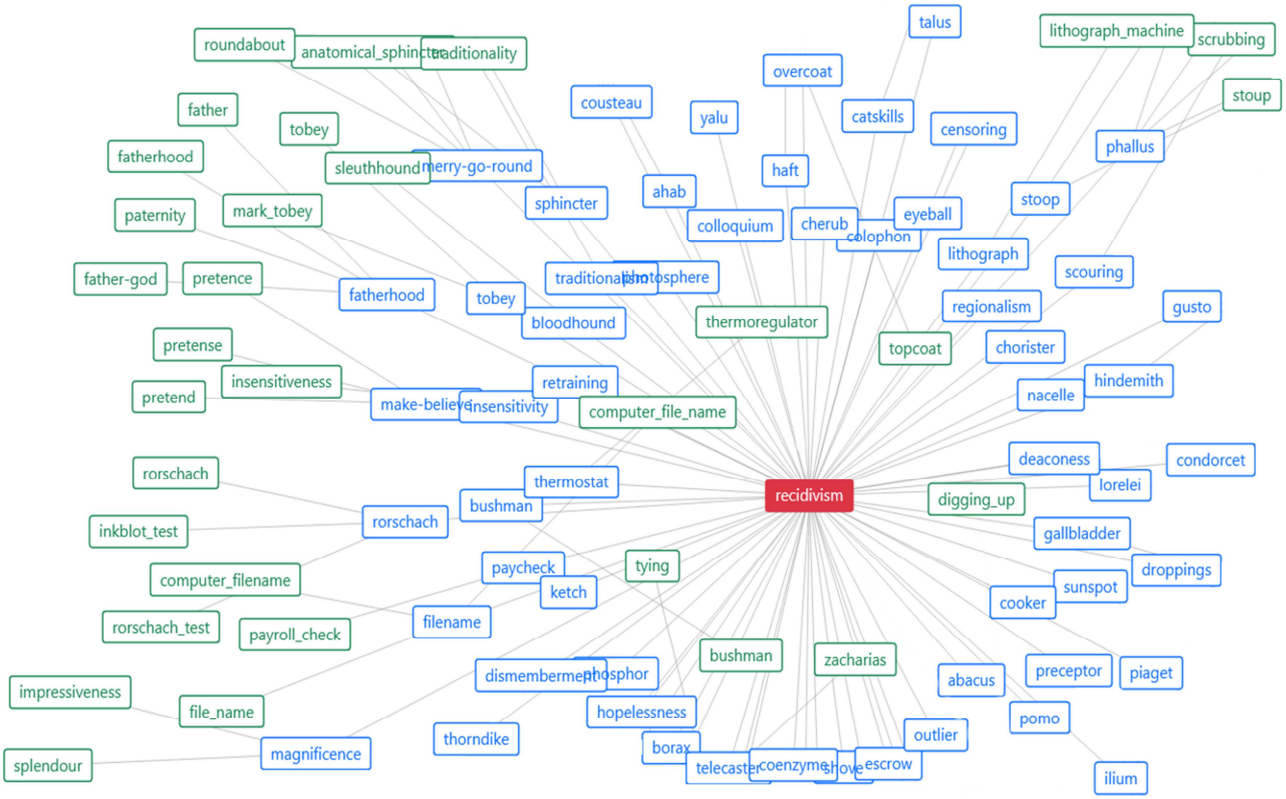

Figure 3. *SN for the initial noun recidivism showing 181 new NCP and SCP nouns for COMPAS – as illustration.*

The acquired new knowledge can be investigated by a human expert in order to mitigate bias, or more comfortably, be added to the training data as feature expansion, and thus, processed automatically. It is worth to note, that the search for new knowledge, i.e. NCP and SCP algorithms, can be iterated until no new nouns will be found.

5. Evaluation Investigating the Autonomous Vehicle Crash Tesla

Autonomous vehicles have severe challenges in public spaces in order to avoid accidents with other vehicles and humans. However, they are one of the big challenges for AI-based for autonomous vehicles [33]. Many systems for autonomous car driving are relying on camera input and the corresponding automatic scene analysis. A famous example is the Tesla car accident that happened 2016 [26]. Unfortunately, an autonomous driving Tesla car had a fatal crash with a van that was not recognized by the AI-based scene analysis and interpretation system of Tesla's camera and sensor input. The applied neural networks for the scene interpretation might have been incomplete for the existent scene, i.e. the training dataset didn't cover enough examples for white vans in order to discriminate them from the sky.

An explanation is provided from the University of Southampton transportation research group as the result of the investigation [26]: "*Although the NHTSA (2017) report concludes that there were no functional problems that led to the subsequent accident, there is speculation that the radar and camera technology failed to detect the trailer against a brightly lit sky or that the trailer was misclassified as an overhead sign by the software. Regardless, the driver would have detected the conflict between what the HMI was showing, how the vehicle was behaving and the information that was actually available in the real world. At this point, the driver would have resumed manual control of the vehicle and the outcome may have been different.*" However, the investigators express that the explanation is uncertain. We investigate in the following this explanation with the NCP and SCP approach. The above cited FTB contains 20 nouns: report, problem, accident, speculation, radar, camera, technology, trailer, sky, sign, software, driver, conflict, vehicle, information, world, point, control, vehicle, and outcome. TDIBM found 135 FOS nouns in the NCP and in the SCP 588 SOS nouns. For a better explanation the noun *control* is highlighted.

Figure 4 shows the SN for the starting noun *control*. Among the found knowledge that consists of 135 + 588 = 723 nouns, the noun *dominance* (of the driver) describes how strong a driver trusts the autonomous driving system or with other words, how strong s/he wants to stay in control. Among other interesting nouns, in the NCP the noun *command* (blue word circled) has been found and then in the SCP the noun *bidding* (green word circled). This raises and implies the idea that the computer vision system should be able to perform biddings: Assume all cameras and sensor systems exist threefold. Since the

sensors have physically different positions and thus, they will differ in their viewing angles for the scene, different inputs for the AI-based scene interpretation system are expected and they will lead to different interpretations. In this case the threefold interpretations can be used for a bidding systems that favors the most possible interpretation for the measured scene. In the forehand scene would this be the forced control and command for the driver with the expectation that "*the driver would have resumed manual control of the vehicle and the outcome may have been different" [26]*. The bidding that is proposed by TDIBM may solve erroneous scene interpretations and prevent fatal accidents by autonomous driving vehicles.

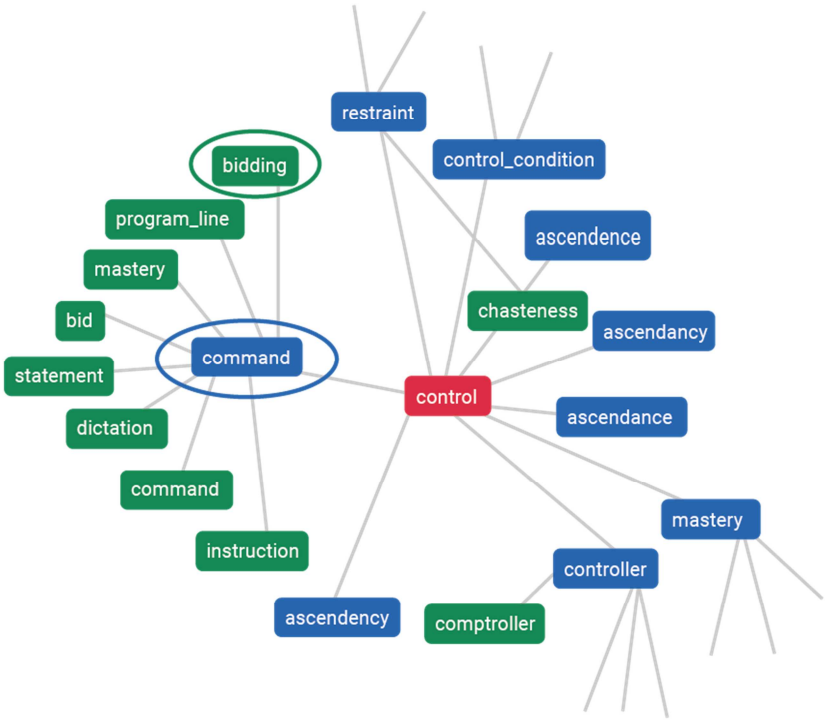

Figure 4. *SN showing NCP and SCP for Tesla with the starting noun control (red).*

6. Conclusion

The new approach for bias mitigation TDIBM minimizes the challenging semantic bias, where missing training features are unknown or domains are too complex for human experts. As a consequence, the search goal is unknown and cannot be described. Human involvement of an expert that is in many cases necessary is reduced to a minimum and can be solved in the optimal case fully automatic by adding the newly found knowledge as features to the training data for an AI system. Two evaluations COMPAS and Tesla prove the effectiveness of TDIBM: In both real use cases new valuable knowledge has been found that consists for COMPAS of 181 previously unknown nouns and for Tesla of 723 entities. TDIBM is a new approach for the case where the search target cannot be described or is difficult to describe (e.g., missing knowledge in the training domain), which collects missing training knowledge from the real world starting from a domain for AI systems.

As next steps stemming can be applied to TDIBM with NCP and SCP. Stemming is the expansion of considering nouns and their related verbs [12]. Analogously, lemmatizing can be investigated, too.

Acknowledgements

Kai Gutberlet, Justin Toffel, and Martin Ziegler are acknowledged for implementing the algorithms and for valuable comments on the algorithm design.

References

[1] Morik, K., Kietz, B., Emde, W., & Wrobel, S. (1993). *Knowledge acquisition and machine learning*. Morgan Kaufmann Publishers Inc.

[2] Lloyd, J. W. (2012). *Foundations of logic programming*. Springer Science & Business Media.

[3] Goñi, J., Arrondo, G., Sepulcre, J., Martincorena, I., de Mendizábal, N. V., Corominas-Murtra, B., & Villoslada, P. (2011). The semantic organization of the animal category: evidence from semantic verbal fluency and network theory. *Cognitive processing*, *12* (2), 183-196.

[4] Shapiro, S. C., & Rapaport, W. J. (1987). SMePS considered as a fully intensional propositional semantic network. In *The knowledge frontier* (pp. 262-315). Springer, New York, USA.

[5] Heydon, A., & Najork, M. (1999). Mercator: A scalable, extensible web crawler. *World Wide Web*, *2* (4), 219-229.

[6] Boldi, P., Codenotti, B., Santini, M., & Vigna, S. (2004). Ubicrawler: A scalable fully distributed web crawler. *Software: Practice and Experience*, *34* (8), 711-726.

[7] Shkapenyuk, V., & Suel, T. (2002). Design and implementation of a high-performance distributed web crawler. In *Proceedings 18th International Conference on Data Engineering,* pp. 357-368. IEEE.

[8] Thelwall, M. (2001). A web crawler design for data mining. *Journal of Information Science*, *27* (5), 319-325.

[9] Kusner, M., Sun, Y., Kolkin, N., & Weinberger, K. (2015). From word embeddings to document distances. In *International conference on machine learning*, pp. 957-966.

[10] Lin, D. (1998). An information-theoretic definition of similarity. In *ICML*, vol. 98, pp. 296-304.

[11] Islam, A., & Inkpen, D. (2008). Semantic text similarity using corpus-based word similarity and string similarity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *2* (2), 1-25.

[12] Song, W., Feng, M., Gu, N., & Wenyin, L. (2007). Question similarity calculation for FAQ answering. In *Third International Conference on Semantics, Knowledge and Grid (SKG 2007)*, pp. 298-301. IEEE. And http://wordnet.princeton.edu visited on January 2022.

[13] Englert, R., & Muschiol, J. (2020). Syntactic and Semantic Bias Detection and Countermeasures. In *International Conference on Computational Science*, pp. 629-638. Springer, Cham.

[14] Neff, G., & Nagy, P. (2016). Automation, algorithms, and politics| talking to Bots: Symbiotic agency and the case of Tay. *International Journal of Communication*, *10*, 17.

[15] Angwin, J., Larson, J., Mattu, S., & Kirchner, L (2016). Machine bias. Pro Publica, May 23.

[16] Zimmermann, J. & Cremers, A. (2019). Foundations of Artificial Intelligence and Effective Universal Induction. In *PAS-PASS Conference, Robotics, Artificial Intelligence and Humanity: Science, Ethics and Policy.*

[17] Englert R. (1998). Learning Model Knowledge for 3D Building Reconstruction. PhD thesis. Bonn University. Faculty III. Germany.

[18] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv: 1301.3781*.

[19] Rada, R., Mili, H., Bichnell, E., & Blettner, M. (1989). Development and Application of Metric on Semantic Nets, *IEEE Transactions on Systems, Man, and Cybernetics*, 9 (1): 17-30.

[20] Lafferty, J., Sleator, D., & Temperley, D. (1992). *Grammatical trigrams: A probabilistic model of link grammar* (Vol. 56). School of Computer Science, Carnegie Mellon University. And https://www.link.cs.cmu.edu/link/index.html visited on January 2022.

[21] Stevenson, A. (Ed.). (2010). *Oxford dictionary of English*. Oxford University Press, USA.

[22] Broder, A. (2000). Identifying and filtering near-duplicate documents. In *Annual Symposium on Combinatorial Pattern Matching*, pp. 1-10. Springer, Berlin, Heidelberg.

[23] Leech, G., & Rayson, P. (2014). *Word frequencies in written and spoken English: Based on the British National Corpus*. Taylor & Francis, Routledge.

[24] Brennan, T., Dieterich, W., & Ehret, B. (2008): Evaluating the Predictive Validity of the Compas Risk and Needs Assessment. http://cjb.sagepub.com/cgi/content/abstract/36/1/21. On behalf of: International Association for Correctional and Forensic Psychology.

[25] Dressel, J., & Farid, H. (2018). The accuracy, fairness, and limits of predicting recidivism. *Science advances*, *4* (1), eaao5580.

[26] Banks, V. A., Plant, K. L., & Stanton, N. A. (2018). Driver error or designer error: Using the Perceptual Cycle Model to explore the circumstances surrounding the fatal Tesla crash on 7th May 2016. *Safety science*, *108*, 278-285.

[27] https://www.english-corpora.org/iweb/, visited July 2021.

[28] Bias in the face recognition Software of Google. https://www.bbc.com/news/technology-45561955, visited December 2021.

[29] Oxford dictionary and similar words, https://www.oxfordlearnersdictionaries.com/definition/english/skin_1?q=skin, visited December 2021.

[30] Spärck, J. K. (1972). "A Statistical Interpretation of Term Specificity and Its Application in Retrieval". Journal of Documentation. 28: 11–21. CiteSeerX 10.1.1.115.8343. doi: 10.1108/eb026526.

[31] Semenov, I., & Arefin, S. (2019). English word frequencies from all English Wikipedia articles. https://github.com/IlyaSemenov/wikipedia-word-frequency, visited December 2021.

[32] Fellbaum, C. (2005). WordNet and wordnets. In: Brown, Keith et al. (eds.), Encyclopedia of Language and Linguistics, Second Edition, Oxford: Elsevier, 665-670.

[33] Levinson, J., Askeland, J., Becker, J., Dolson, J., Held, D., Kammel, S.,... & Thrun, S. (2011). Towards fully autonomous driving: Systems and algorithms. In *2011 IEEE intelligent vehicles symposium (IV)*, 163-168.

[34] Leech, G., Garside, R., & Bryant, M. (1994). CLAWS4: the tagging of the British National Corpus. In *COLING 1994 Volume 1: The 15th International Conference on Computational Linguistics*.