Automatic Relation Extraction for Building Smart City Ecosystems using Dependency Parsing*

Daniel Braun, Anne Faber, Adrian Hernandez-Mendez, and Florian Matthes

Department of Informatics, Technical University of Munich, Munich, Germany
{daniel.braun,anne.faber,adrian.hernandez,matthes}@tum.de
https://wwwmatthes.in.tum.de/

Abstract. Understanding and analysing rapidly changing and growing business ecosystems, like smart city and mobility ecosystems, becomes increasingly difficult. However, the understanding of these ecosystems is the key to being successful for all involved parties, like companies and public institutions. Modern Natural Language Processing technologies can help to automatically identify and extract relevant information from sources like online news and blog articles and hence support the analysis of complex ecosystems. In this paper, we present an approach to automatically extract directed relations between entities within business ecosystems from online news and blog articles by using dependency parsing.

Keywords: Relation Extraction · Dependency Parsing · Smart City · Ecosystem.

1 Introduction

Digitization - and its advancements - has long reached cities including their outskirts and rural satellites and is changing urban mobility. Cities are transforming into Smart Cities, whereby Smart Mobility is often recognized among the most common indicators of Smart Cities [4]. Digital technologies are continuously integrated in vehicles, traffic systems, and infrastructure [16] and are thereby changing the mobility demands of humans. The variety of digital technologies range from mobile applications to Internet of Things (IoT) devices integrated in existing infrastructure. Thereby, mobility applications provide timely information on the traffic situation, the option to buy tickets for public transportation

* This work is part of the TUM Living Lab Connected Mobility (TUM LLCM) project and has been funded by the Bavarian Ministry of Economic Affairs, Energy and Technology (StMWi) through the Center Digitisation.Bavaria, an initiative of the Bavarian State Government.
This work has been sponsored by the German Federal Ministry of Education and Research (BMBF) grant BEEx+ 01IS17049.
This work has been sponsored by the German Federal Ministry of Education and Research (BMBF) grant A-SUM 01IS17049.

online, or the usage of shared mobility services such as car sharing, bike sharing or ride sharing, to name just a few. IoT devices such as sensors make information of occupied or free parking slots available or report about the carbon dioxide (CO2) level on roads with heavy traffic [19].

Established mobility actors, such as automotive OEMs, their Tier 1 to 3 parts supplier, but also public transportation agencies, are challenged especially by technology companies using their advantage of applying new technologies - such as augmented reality or artificial intelligence - to urban mobility. Tech giants such as Google and Apple are entering the mobility scene by developing self-driving cars and pushing autonomous driving [5, 20] exhibiting disruptive innovative characteristics. Thus, new actors enter and transform the existing mobility markets that are geographically focused on specific metropolitan areas. As a result, new mobility business ecosystems are currently emerging. With new technologies being used and applied, also mobility related legislation has to be discussed and adapted, turning cities, public institutions and their governments into actors of these ecosystems.

Besides commercial mobility providers, also cities, their public institutions, and their governments are under pressure to address these challenges and to understand the emerging structures within mobility ecosystems to make informed decisions [10].

Thereby, ecosystem data is large and heterogeneous [2], ranging from technology-related data about applied standards and platforms to use patterns of mobility service apps and their user types. When focusing on the business aspects of these emerging mobility ecosystems, information about service providers, their strategies, partnerships and offered solutions, and cooperative initiatives become relevant [6]. Data comprising this information can come from various sources, such as existing databases of the established mobility ecosystems, newspaper articles or blogs addressing recent development within the ecosystem, but also company and institutional web presences and publications. Few research has looked into the issues related to data collection in emergent business ecosystems [9, 8].

The manual collection and extraction of the data can be considered as a time-consuming and tedious work providing a noticeable limitation for the data-driven analysis of the business ecosystem. An automation of this process could not only save valuable resources but could also enable (almost) real-time availability of the data and hence create possibilities for more advanced analyses of changes within an ecosystem and foster more advanced Artificial Intelligence (AI) systems, which could not only be useful for actors within the ecosystem, but also e.g. for financial analysts.

The here presented research is part of a smart city initiative pursued by a European city. Within this paper, we present an approach to automatically extract directed relations about actors within smart city ecosystems from internet news and blog articles by using dependency trees. We present and evaluate a prototypical implementation of this approach, combining machine learning methods (for dependency parsing) and rule-based approaches (for relation di-

rection extraction). Such a system could in the future be used in combination with visualisation tools, in order to foster the manual analysis of such ecosystems by humans, or with AI systems, in order to enable a more automated and data-driven analysis.

2 Related Work

A very popular field of application for relation extraction is bioinformatics, where the relation between genes and proteins is extracted from scientific publication. Often, kernel methods are used to achieve this goal, e.g. by [17], [14], [1], and more recently [18].

Lee et al. [13] used convolutional neural networks to extract the directed relations "hypnoym of" and "synonym of" from scholarly articles. In their evaluation, they achieved an F1 score of 0.645.

Fundel et al. [7] presented an approach which is very similar to the approach we present in this paper. They used named entity recognition (NER) and dependency trees in order to extract relations. However, in the domain of proteins and genes, NER is a much easier task compared to names of companies and institutions, which often consist of "regular" words. Moreover, they just extracted two types of relations: A activates B (and its inversion) and A interacts with B. In smart city ecosystems, there are much more different types of relation which could be of interest.

Yamamoto et al. [21] used the DeepDive system [22] to extract relations between companies in the semiconductor industry from web news articles. However, they only focused on undirected, high-level relations: collaboration and competition. The same two relations were also extracted by Lau and Zhang [12] by using a Support Vector Machine. While they achieved an F1-score of 0.868 when it comes to business entity identification, for the relation extraction they only achieved an F1-score of 0.625 (collaboration) respectively 0.631 (competition).

Both approaches work with English texts. In contrast, we work with texts in German language and want to extract more fine-grained, directed relations.

3 Dataset

In this paper, we want to focus on three important relations which describe the constitution of smart city ecosystems and more broadly business ecosystem in general: "owns", "funds", and "cooperation".

While the first two of these relations are directed (i.e. A owns B $\not\Longrightarrow$ B owns A), the relation cooperation is not directed (i.e. A cooperation B $\Longrightarrow$ B cooperation A). In order to evaluate our approach, we manually collected a dataset of 41 news and blog articles that contain information about one of the above-mentioned relation between two companies from the smart city ecosystem and manually annotated it on a document level. The sources from which the articles were extracted include major German news outlets (like "Welt"[1] or

[1] https://www.welt.de

"Handelsblatt"[2]) as well as small niche blogs (like "eMobilitaet"[3]). Every article was annotated by two persons with a consensual annotation. Figure 1 shows the distribution of the three relations within the collected dataset. Table 1 shows the companies which are included in the dataset along with the number of relations they occur in.

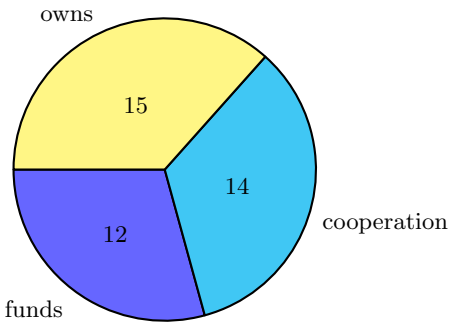

Fig. 1. Number of occurrences of the different relations within the dataset

4 Approach

In order to automatically extract relations from online news and blog articles, a set of preprocessing steps has to be conducted first. For this, we use a pipes and filters architecture [15], which is shown in Figure 2. In a first step, we extract the main article content from the website by removing unrelated elements like the header, navigation or footer by using the *boilerpipe* Java library[4], which uses shallow text features to separate the main content from the other structure. [11] Subsequently, HTML tags are stripped from the main content and the text is split into sentences. These sentences are the basis for the detection and extraction of the relations.

In order to avoid having to annotate huge sets of training data for each and every type of relation, we decided to use a hybrid approach, combining machine learning and manually crafted rules. We use the dependency parser developed by Chen and Manning [3], which uses neural networks to create dependency trees for multiple languages, including German.

We then created rules which extract, based on the dependency trees, the relations we are interested in. Figure 3 shows, for example, the dependency graph for the sentence "Daimler und BMW kooperieren für verbessertes Carsharing-Erlebnis." (*Daimler and BMW cooperate for improved Carsharing-experience*).

[2] https://www.handelsblatt.de
[3] https://www.emobilitaetblog.de
[4] https://boilerpipe-web.appspot.com/

company	#	company	#	company	#
ADAC	1	Grab	1	SolarCity	1
Albertini Cesare	1	Harley Davidson	1	StoreDot	1
Alta Motors	1	IOTA	1	Streetscooter	1
Amazon	1	Infineon	1	Tass	1
Apple	1	Innoviz	1	Telekom	1
Aurora	1	Jaguar	1	Tencent	1
BMW	8	Jump Bikes	1	Tesla	3
BP	1	Meituan Dianping	1	Toyota	1
Blacklane	1	Mobike	1	Uber	1
Bosch	6	Momenta	1	VeChain	1
Byton	1	NewMotion	1	Volkswagen	3
Chauffeur Prive	1	Nissan	1	Waymo	1
CleverShuttle	1	Perbix	1	What3Words	1
Cobi	1	Porsche	1	Xain	1
Coup	1	QuantumScape	1	car2go	1
Daimler	9	Relayr	1	flinc	1
DriveNow	1	Seeo Inc.	1	tiramizoo	1
E.ON	1	Shell	2		
Escrypt	1	Siemens	1		
FAW	1	Snapchat	1		

Table 1. Companies within the dataset and how often they occur

In this sentence, we find the undirected relation "cooperation" between the entities Daimler and BMW. In order to automatically detect relations, we first check for each sentence whether it includes information about one of the relations we are looking for (owns, funds, cooperation). In order to do this, we define one or more keyword for each of the relations: e.g. "kooperiert" for cooperation, "investieren" for funds, and "übernehmen" and "kauft" for owns. We stem each sentence using the Snowball stemmer[5] and subsequently search for occurrences of the (stemmed) keywords or synonyms of them which we acquire through the Open Thesaurus API[6].

Another advantage of our (partially) rule-based approach is the fact that it can easily be transferred to other languages by translating the identified keywords and using a different Thesaurus and model for the dependency parser.

In the case of the example in Figure 3, once we identified the token which defines the type of the relation ("kooperieren"), it is sufficient to look for the Named Entities (NE), i.e. nouns, which are subject to this token, hence we are looking for `nsubj` connections in the graph, which leads us to the tokens BMW and Daimler.

For the two other relations, owns and funds, rules can get a bit more complex, since both relations are directed and it is therefore not sufficient to just

[5] http://snowballstem.org/
[6] https://openthesaurus.de

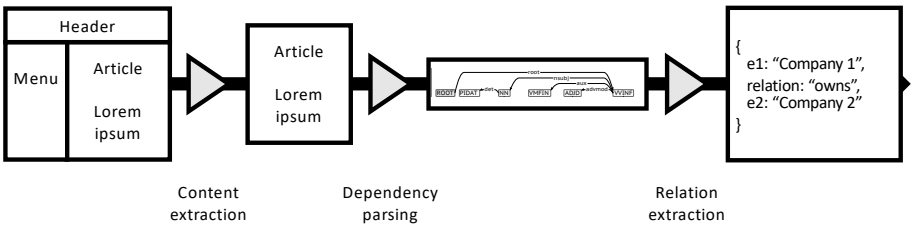

Fig. 2. Pipes and filters architecture of the prototype

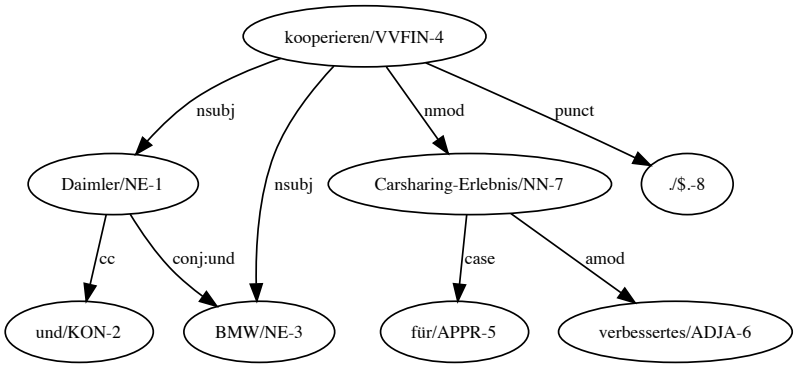

Fig. 3. Dependency graph for the sentence "Daimler und BMW kooperieren für verbessertes Carsharing-Erlebnis." (*Daimler and BMW cooperate for improved Carsharing-experience.*)

identify the related entities, but it is also necessary to detect and extract the direction of the relation. The two sentences "Siemens übernimmt niederländisches Startup Tass." (*Siemens acquires Dutch startup Tass.*) and "Niederländisches Startup Tass wird von Siemens übernommen." (*Dutch startup Tass is acquired by Siemens.*) are basically identical, but one time the German verb "übernehmen" (to acquire) is used in its active form and once in the passive form. However, this small difference is very important for the direction of the relation.

One of the reasons why we choose an approach using dependency trees is their power when it comes to distinguishing the direction of a relation, as shown in Figure 4 and 5. In Figure 4, "Siemens" is the nominal subject (`nsubj`), hence, in the relation, "Siemens" is the company which acquired another company. Therefore, we just have to look for the second Named Entity in the sentence to get the full relation. In Figure 5, "Tass" is the **passive** nominal subject

(`nsubjpass`), hence, it is clear that "Tass" is the company which **was** acquired by another company.

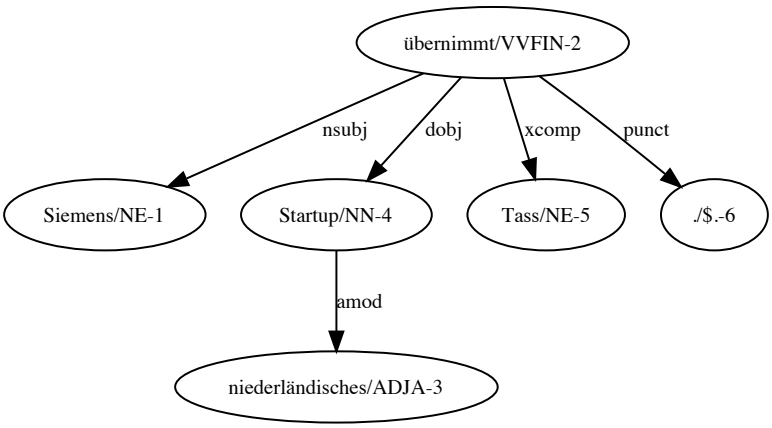

Fig. 4. Dependency graph for the sentence "Siemens übernimmt niederländisches Startup Tass." (*Siemens acquires Dutch startup Tass.*)

The same rules apply for the "funds" relation. In general, these are obviously just examples and not an exhaustive set of rules. The keyword can, for example, not only occur as a verb, like in the examples we gave before, but also in form of a noun, as shown in Figure 6 and Figure 7. We can again distinguish the direction of the relationship by looking at the nominal subject (`nsubj`, Figure 6) or **passive** nominal subject (`nsubjpass`, Figure 7) relation.

The rules we use in our prototype were developed based on an existing, handcrafted, database which contains more than 470 structured relations between companies from the smart city domain and links to the websites the information was (manually) extracted from. This database is distinct from the set described in Section 3 which we will use to evaluate our prototype. In addition to the three relations we focus on (owns, funds and cooperation), this database also includes the additional relations "supplied by" and "partially owns".

In order to make our prototype universally applicable and easy to integrate with existing tools, we decided to provide the functionality through a REST API. The prototype itself is, therefore, a standalone Java application.

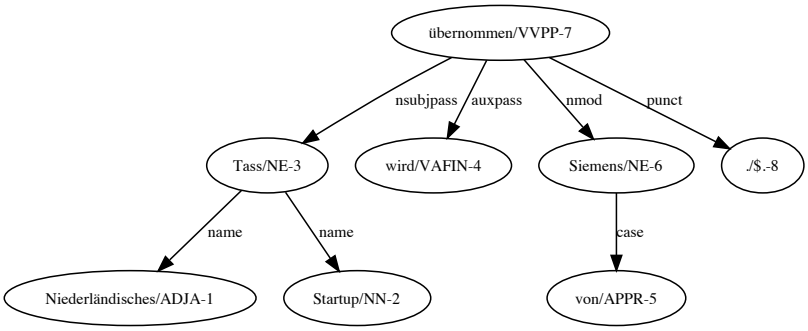

Fig. 5. Dependency graph for the sentence "Niederländisches Startup Tass wird von Siemens übernommen." (*Dutch startup Tass is acquired by Siemens.*)

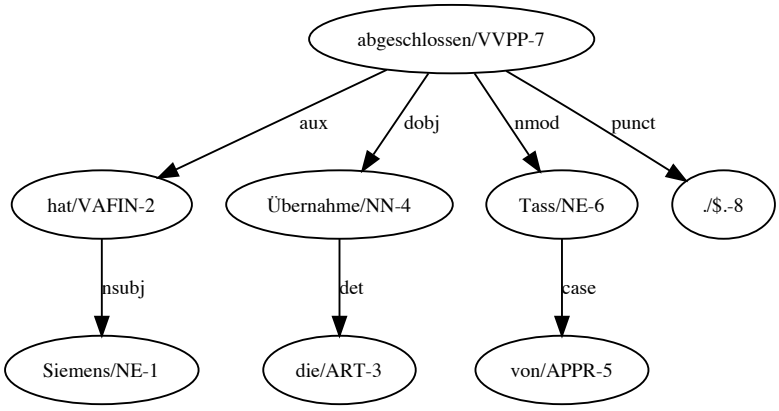

Fig. 6. Dependency graph for the sentence "Siemens hat die bernahme von Tass abgeschlossen." (*Siemens completed the acquisition of Tass.*)

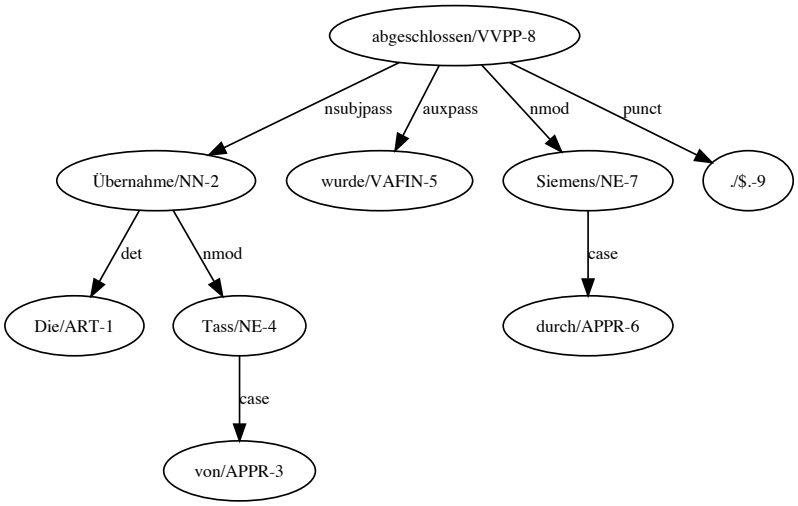

Fig. 7. Dependency graph for the sentence "Die bernahme von Tass wurde durch Siemens abgeschlossen." (*The acquisition of Tass was completed by Siemens.*)

5 Limitations

While the usage of manually crafted rules has some advantages, it also introduces some limitations. The success of the system is for example highly dependent on the quality of the thesaurus which is used. In this paper, we only look at a very small subset of relations which might be of interests for smart city ecosystems. Especially differentiation between relations like "owns" and "partially owns" might prove to be difficult with the approach we choose. For some relations, it might even for humans be difficult to distinguish between them and not all relations are necessarily exclusive, like e.g. "supplied by" and "cooperates". Again, since we only look at a small subset of relations, we did not encounter any of these problems for our prototype.

6 Evaluation

In order to evaluate our approach, we used the dataset described in Section 3 with our prototype. We evaluated both, the extraction of the relation itself (i.e. whether the relations "owns", "finances" or "cooperates" are extracted correctly) and the evaluation of the involved entities. For the directed relations, we also evaluated whether the direction of the relations was extracted correctly.

The results of the evaluation of the relation extraction are shown in Table 2. Overall, with an F1 score of 0.862 (unweighted average) the results look very promising, especially given the fact that we used a general-purpose thesaurus without any content specialised for the task we wanted to solve, and significantly outperform e.g. the results achieved by [12].

class	instances	cooperates	funds	owns	none	precision	recall	F1 score
cooperates	14	12	0	2	0	0.8	0.857	0.828
funds	12	1	11	0	0	1	0.917	0.957
owns	15	2	0	12	1	0.857	0.8	0.828
Σ	41	15	11	14	1	0.867	0.858	0.862

Table 2. Results of the evaluation of the relation extraction

However, the relation extraction alone is not yet very meaningful. For the task we want to solve, it is also crucial that the right entities are extracted and, for the directed relations, also that they are extracted in the right order. For this evaluation, we were not only considering exact matches as correct but also variations, like "VW" for "Volkswagen" or "Infineon" for "Infineon Technologies". In production-use, these variations would need to be mapped in order to be unambiguous.

The results of this evaluation are shown in Table 3. Each of our relations contains two entities. In the evaluation, we distinguish whether no entity, one entity or both entities were extracted correctly, independent from their direction. Only in the column "correct direction" we distinguish whether the extracted direction was correct or not. If just one entity was extracted correctly, the direction is considered to be correct if this one entity is on the "right side" of the relation. For this evaluation, we only took into account the true positives identified in the previous evaluation.

class	instances	no entity	one entity	two entity	correct direction
cooperates	12	0	6	6	n.a.
funds	11	1	1	9	10
owns	12	3	5	4	8
Σ	35	4	12	19	18

Table 3. Results of the evaluation of the entity extraction and relation direction

Overall, the 35 correctly classified relations contained 70 entities. Out of this 70 entities, our prototype correctly extracted 50 entities and failed to extract 20 entities. During the evaluation, it was obvious that the standard NER we used from the Stanford Library is not optimal for the task. Even big company names like "Volkwagen" were, in some cases, not recognised as named entities. However,

only in four out of 35 cases none of the involved entities could be extracted. In cases were at least one entity could be extracted, the direction of the relation was correctly extracted in 18 out of 19 cases.

Overall, our prototype performed very well, especially with regard to the extraction of relations and their direction, which distinguishes our prototype from most of the approaches presented in Section 2, which do not consider the direction of a relation.

7 Conclusion

In this paper, we presented an approach using dependency trees to automatically extract directed relations between companies from German news and blog articles in order to automatically analyse smart city ecosystems. With an F1 score of 0.862, our prototype was successful in detecting relations and with 94.74% correctly directed relations even more so with regard to the direction of detected relations. Currently, the main shortcoming of the system is the extraction of involved entities. While we were able to extract at least one of the involved entities correctly in 88.57% of the cases, both entities were extracted correctly only in 54.29% of all cases. In the future, this value could be improved by using more sophisticated methods for named entity recognition.

In the future, our prototype could be combined with visualisation tools, in order to foster the manual analysis of such ecosystems by humans, or with AI systems, in order to enable a more automated and data-driven analysis, using the REST API we implemented.

References

1. Airola, A., Pyysalo, S., Björne, J., Pahikkala, T., Ginter, F., Salakoski, T.: All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning. BMC bioinformatics **9**(11), S2 (2008)
2. Basole, R.C., Russell, M.G., Huhtamäki, J., Rubens, N., Still, K., Park, H.: Understanding business ecosystem dynamics: A data-driven approach. ACM Transactions on Management Information Systems (TMIS) **6**(2), 6 (2015)
3. Chen, D., Manning, C.: A fast and accurate dependency parser using neural networks. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 740–750 (2014)
4. Chourabi, H., Nam, T., Walker, S., Gil-Garcia, J.R., Mellouli, S., Nahon, K., Pardo, T.A., Scholl, H.J.: Understanding smart cities: An integrative framework. In: System Science (HICSS), 2012 45th Hawaii International Conference on. pp. 2289–2297. IEEE (2012)
5. Etherington, D.: Googles self-driving car unit becomes waymo. accessed: 16.04.2018 (2016), https://techcrunch.com/2016/12/13/googles-self-driving-car-unit-spins-out-as-waymo/
6. Faber, A., Hernandez-Mendez, A., Rehm, S.V., Matthes, F.: An agile framework for modeling smart city business ecosystems. In: Proceedings of the 20th International Conference on Enterprise Information Systems, vol. 2. pp. 39–50 (2018)

7. Fundel, K., Kffner, R., Zimmer, R.: Relexrelation extraction using dependency parse trees. Bioinformatics **23**(3), 365–371 (2007). https://doi.org/10.1093/bioinformatics/btl616, http://dx.doi.org/10.1093/bioinformatics/btl616
8. Hao, J., Zhu, J., Zhong, R.: The rise of big data on urban studies and planning practices in china: Review and open research issues. Journal of Urban Management **4**(2), 92–124 (2015)
9. Iyer, B.R., Basole, R.C.: Visualization to understand ecosystems. Communications of the ACM **59**(11), 27–30 (2016)
10. Khatoun, R., Zeadally, S.: Smart cities: concepts, architectures, research opportunities. Communications of the ACM **59**(8), 46–57 (2016)
11. Kohlschütter, C., Fankhauser, P., Nejdl, W.: Boilerplate detection using shallow text features. In: Proceedings of the third ACM international conference on Web search and data mining. pp. 441–450. ACM (2010)
12. Lau, R., Zhang, W.: Semi-supervised statistical inference for business entities extraction and business relations discovery. Balog et al.[3] pp. 41–46 (2011)
13. Lee, J.Y., Dernoncourt, F., Szolovits, P.: Mit at semeval-2017 task 10: relation extraction with convolutional neural networks. arXiv preprint arXiv:1704.01523 (2017)
14. Li, J., Zhang, Z., Li, X., Chen, H.: Kernel-based learning for biomedical relation extraction. Journal of the Association for Information Science and Technology **59**(5), 756–769 (2008)
15. Meunier, R.: The pipes and filters architecture. In: Pattern languages of program design. pp. 427–440. ACM Press/Addison-Wesley Publishing Co. (1995)
16. Mitchell, W.J., Borroni-Bird, C.E., Burns, L.D.: Reinventing the automobile: Personal urban mobility for the 21st century. MIT press (2010)
17. Mooney, R.J., Bunescu, R.C.: Subsequence kernels for relation extraction. In: Advances in neural information processing systems. pp. 171–178 (2006)
18. Panyam, N.C., Verspoor, K., Cohn, T., Ramamohanarao, K.: Exploiting graph kernels for high performance biomedical relation extraction. Journal of biomedical semantics **9**(1), 7 (2018)
19. Tapashetti, A., Vegiraju, D., Ogunfunmi, T.: Iot-enabled air quality monitoring device: A low cost smart health solution. In: Global Humanitarian Technology Conference (GHTC), 2016. pp. 682–685. IEEE (2016)
20. Taylor, M.: Apple confirms it is working on self-driving cars. accessed: 16.04.2018 (2016), https://www.theguardian.com/technology/2016/dec/04/apple-confirms-it-is-working-on-self-driving-cars
21. Yamamoto, A., Miyamura, Y., Nakata, K., Okamoto, M.: Company relation extraction from web news articles for analyzing industry structure. In: Semantic Computing (ICSC), 2017 IEEE 11th International Conference on. pp. 89–92. IEEE (2017)
22. Zhang, C.: DeepDive: a data management system for automatic knowledge base construction. Ph.D. thesis, The University of Wisconsin-Madison (2015)