HPI Hasso
Plattner
Institut
IT Systems Engineering
Universität Potsdam

Bachelor Thesis

Feature Extraction for Business Entity Linking in Newspaper Articles

Merkmalsextraktion zur Unternehmenserkennung in Zeitungsartikeln

Jonathan Janetzki
jonjanetzki@gmail.com

July 21, 2017

Information Systems Group

Supervisors
Prof. Dr. Felix Naumann
Toni Grütze
Michael Loster

Abstract

German newspaper articles contain a lot of recent information about business relations. Their automated retrieval allows to construct the *German Corporate Graph* and keep it up to date. A premise for the relation extraction is NER and EL to identify the mentioned businesses. Since business aliases may be ambiguous, this is a complex problem. Its solution requires the extraction and comparison of significant features for business entity linking.

This thesis comprises a software system that finds and disambiguates references to organizations in newspaper articles using appropriate features. It uses the German Wikipedia that contains explicit link annotations to named entities and learns from it to recognize mentions of organizations. The extracted features are statistical measures, linguistic properties of the alias' context and second order features. The system then applies these features to newspaper articles without link annotations, which allows to find business aliases and their referenced entities.

The distributions of the features' values reveals that they strongly depend on whether a reference is valid or not. This means that the features have a high quality and are applicable by a classifier. Furthermore, the software is scalable in order to be also suitable for economical use on large amounts of data.

Zusammenfassung

Deutsche Zeitungsartikel enthalten eine Menge aktueller Informationen über Unternehmensbeziehungen. Ihre automatische Gewinnung ermöglicht es, den *Deutschen Unternehmensgraphen* zu konstruieren und auf dem neusten Stand zu halten. Eine Voraussetzung zur Relationsextraktion ist NER und EL, um die genannten Unternehmen zu finden. Da Unternehmensnamen mehrdeutig sein können, ist dies eine komplexe Aufgabe. Ihre Lösung erfordert die Extraktion und den Verleich von aussagekräftigen Merkmalen zur Unternehmenserkennung.

Diese Forschungsarbeit umfasst ein Softwaresystem, das Verweise auf Organisationen in Zeitungsartikeln mithilfe von angemessenen Merkmalen eindeutig ermittelt. Es benutzt die deutschsprachige Wikipedia, die explizite Linkannotationen auf benannte Entitäten enthält, und lernt daraus, Nennungen von Organisationen wiederzuerkennen. Die extrahierten Merkmale sind statistische Maße, linguistische Eigenschaften des Kontexts eines Namens und Merkmale zweiter Art. Das System wendet diese Merkmale dann auf Zeitungsartikel ohne Annotationen an, was das Finden von Unternehmensnamen und ihren referenzierten Entitäten ermöglicht.

Die Verteilung der Merkmalswerte zeigt, dass diese stark davon abhängen, ob eine Referenz gültig ist oder nicht. Das heißt, dass die Merkmale eine hohe Qualität besitzen und von einem Klassifikator verwendet werden können. Darüber hinaus ist die Software skalierbar, um auch für große Datenmengen wirtschatlich einsetzbar zu sein.

Contents

1 The German Corporate Graph Project[1]

1.1 Why a German Corporate Graph?

When it comes to economic decisions, uncertainty is a critical issue. Following the rational choice theory approach, every market player is constantly trying to maximize his utility and minimize his effort.

Uncertainty can be described as a lack of information of how a market - or herein the full German economic system - is constituted and about the future behavior of the market players. Presuming that every market player is acting on a rational basis, all information regarding his situation, resources, plans, and relations makes the results of his decisions more predictable. In this manner, we can state: The more relevant information a market player gathers about other players in the market or economy, the better the foundation of his decisions is. The broad range of that kind of information can lead to a significant competitive advantage. So it should be in a rational player's interest to collect as much relevant information as possible.

In a connected economy, a lot of those uncertainties lie in the relations between corporations[2]. This became evident in the so called *Abgas-Skandal* or *Dieselgate* of the Volkswagen AG in 2015, wherein a lot of external suppliers spin out of control, from a financial perspective [1, 4]. This happened although most of the suppliers did not take part in the scandal itself. Since there are a lot of other examples like the *Lehmann Brothers bankruptcy* or any other economic shock event, we can state that relations are a significant factor in the economic evaluation of corporations and their financial risks.

Because there are millions of corporations in the German economy[3] and each corporation can potentially hold relations to hundreds or thousands of other corporations, collecting and to overseeing all those relations becomes a complicated matter.

The *German Corporate Graph Project* is one approach to solve this problem. The project's purpose is to extract business entities from multiple structured knowledge bases (e.g. Wikidata and DBpedia), merge them, enrich them with relations extracted from unstructured documents and finally display the graph so that it can be visually explored.

The project consists of a pipeline, which starts with the import and normalization of structured knowledge bases. The next step is the Deduplication, which is the detection

[1] This section was written by Matthias Radscheit [8]

[2] We define as *corporation* any juristic entity that takes part in the German economy. This includes especially businesses but also other entities like public corporations.

[3] The *Federal Bureau of Statistics* notes 3,469,039 businesses in Germany in 2015 [2]. Following our definition of corporations, this number has to be seen as a lower bound for the total number of corporations in Germany.

and fusion of occurrences of the same entity over multiple knowledge bases. These entities form a graph, whose nodes are businesses and whose edges are the relations between them. This graph is then enriched during the Information Extraction. In this step relations between entities are extracted from unstructured documents using Named Entity Recognition, Entity Linking and Relation Extraction.

The results of all these steps can be viewed and curated in the so-called Curation Interface. This is a web-interface, which can be used to control the pipeline itself, view statistical data generated by other pipeline steps and to view and curate the entities and relations of the graph itself. The final graph can be visually explored by using the Corporate Landscape Explorer, which is also a web-interface.

1.2 One project - seven contributions

This thesis is published as a part of a bachelor's project in 2016/2017 at Hasso-Plattner-Institute in Potsdam, Germany. The project's objective was to build the *German Corporate Graph*, like described above, for Germany's corporate landscape. The project lasted ten months and was accompanied by Commerzbank AG, Germany. As part of the process, the project participants published several theses.

See here a list of all published theses within the project's context:

- Pabst explores *Efficient Blocking Strategies on Business Data* [9].
- Löper and Radscheit evaluate duplicate detection in their thesis *Evaluation of Duplicate Detection in the Domain of German Businesses* [8].
- Schneider's thesis is entitled *Evaluation of Business Relation Extraction Methods from Text* [12].
- Janetzki investigates *Feature Extraction for Business Entity Linking in Newspaper Articles.*
- Ehmüller explores the *Evaluation of Entity Linking Models on Business Data* [3].
- *Graph Analysis and Simplification on Business Graphs* is the title of Gruner's thesis [6].
- Strelow investigates *Distributed Business Relations in Apache Cassandra* [13].

2 Ambiguous Business Names[4]

The graph of Germany's corporate landscape consists of businesses as nodes and their relations as edges. Structured knowledge bases, such as Wikidata and DBpedia, provide valuable information about the companies for themselves. However, they lack information about the relations. Unstructured texts like Wikipedia or newspaper articles discuss a lot of those.

We have developed an approach to extract a relation between two businesses if they are mentioned in the same sentence. Then both of these mentions need to be found and linked to the entities representing these businesses. Finding these mentions is called *Named Entity Recognition* (NER). The next step is to identify their meant entity, which is called *Entity Linking* (EL). The eventual extraction the relation between these two businesses from the sentence is called *Relation Extraction* (RE). Our approach combines NER and EL into a single process and reduces it into a classification problem. The combination of NER, EL and RE make up the Information Extraction component of our project.

This thesis focuses on the development of a reliable implementation for combined NER and EL. Since there is no one-to-one relationship between businesses and their names, also called *aliases*, this is a complex task. On the one hand, a business may have multiple aliases. For example, "Deutsche Bahn AG" is commonly abbreviated with "Deutsche Bahn" or simply "DB". On the other hand, aliases are ambiguous, which means that the same alias may refer to different businesses in different contexts. This applies in particular for abbreviations. For instance, "DB" may also refer to "Deutsche Bank AG". Therefore, we extract expressive features that allow us to disambiguate such business aliases and link them to their meant entity.

Sec. 3 covers related work in the field of both NER and EL. Sec. 4 explains how our combined NER and EL approach works before Sec. 5 characterizes which features we use to perform EL. Sec. 6 then describes our knowledge bases and how we extract features from these. After that, Sec. 7 discusses the quality of the features and the scalability of the implementation. Finally, Sec. 8 will summarize the results and which steps may improve the project in the future.

[4] This section was written in collaboration with Jan Ehmüller [3].

3 Related Work

Various research teams have worked on projects, which aimed at automatically linking aliases in natural language text to German businesses as well. The following outline gives an overview about what they already have attained:

Immer [7] has worked on the project previous to this project. Starting from legal names, he researched how it is possible to generate aliases by abbreviating and pruning them. To disambiguate ambiguous aliases, he used two vector spaces: One for the tf-idf values of words occurring in the neighborhood of recognized aliases and another one for words occurring in newspaper articles in general. Our approach also uses tf-idf vector spaces. In contrast to Immer's system, we do not generate aliases, but we retrieve them from the German Wikipedia.

The CohEEL project [5] performs NER and EL on natural language texts for more general purposes than business entity linking. It also implements the features that our system uses. In contrast to this project, we perform NER and EL as one combined process.

Another peculiarity of our approach is the distributed execution through cluster computing. The next sections describes we find business aliases and their corresponding entities.

4 Combined NER and EL in Newspaper Articles

Newspaper articles contain a lot of recent information about business relations. E.g., a German article might include a sentence like: "*Bosch liefert Servomotoren an die DB.*" (English: "Bosch delivers servo motors to DB.")

For a human, it is obvious that this sentence represents a delivery relation between the entities: "Robert Bosch GmbH" and "Deutsche Bahn AG". While "DB" could also abbreviate "Deutsche Bank AG", the machine related context makes clear that the alias refers to the railway company instead of the bank.

As such relations are useful for the German Corporate Graph, we want to extract those from texts, such as newspaper articles. To do so, we need to perform NER and EL beforehand. We have combined both steps to a single process since the EL depends on the results of the NER.

4.1 Newspaper data sources

Currently, we perform combined NER and EL on the German Wikipedia[5] and the German newspaper Spiegel Online[6]. As Wikipedia already contains unambiguous hyperlinks to named entities, we use it as training data (see Sec. 6) and ground truth for the evaluation [3]. Newspaper articles are missing such annotations.

For Spiegel Online, we use a dump that consists of HTML, contains 240,000[7] articles and has a total size of 22 GiB. On average, each article's text consists of 550 tokens, where a token is a semantic element of a text. This is a word in most cases, but can also be a special character, such as a comma.

4.2 Process

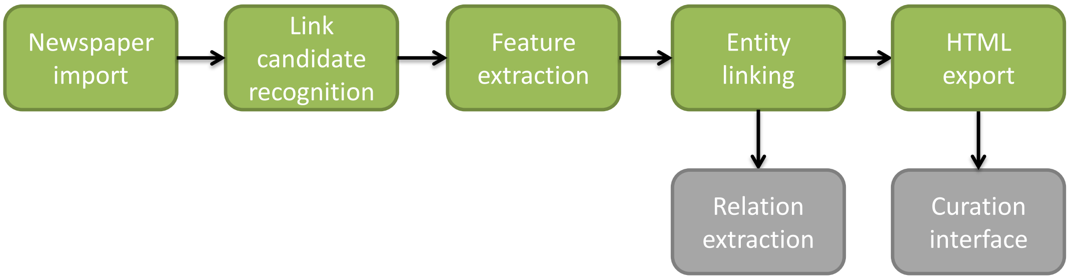


Figure 1: Steps of the combined NER and EL process (green) and the software components building on these (gray)

The combined NER and EL process consists of multiple steps, as depicted in Figure 1. Apart from the data import, the steps are independent of the newspaper source so that we are able to use them for any other as well. The following paragraphs explain the steps in detail.

Newspaper import For each article, the Spiegel Online dump provides the title, the text and a unique internal ID. We extract these three values from the dump and store them in our database.

[5] `https://de.wikipedia.org/`, last accessed on July 21, 2017.
[6] `http://www.spiegel.de/`, last accessed on July 21, 2017.
[7] Here and in the following, numbers are rounded to two significant digits for better readability.

Link candidate recognition A mention of a named entity in a text is a chain of one or more consecutive tokens, which is also called an n-gram for n tokens. The task of named entity recognition is to identify such n-grams. This step is important for our process, as we need to perform EL on a pre-selection of those. Otherwise, it would have to try to link each n-gram in the text to a named entity, which would lead to a computational complexity that increases exponentially with n. This would be unacceptable for economical use.

In contrast to NER, we do not search for named entities, but for n-grams that may signify a named entity depending on their context. This creates a larger pre-selection of *link candidates*. Our EL will, later on, decide to which entity a link candidate refers, which is a very similar task to determining whether it refers to a named entity at all.

We find candidates of named entities simply by selecting all those n-grams that were named entities of organizations within the training data. An efficient way to match those in texts is using a trie that we have constructed during the preprocessing of the raw data (see Sec. 6.2, Alias analysis).

Feature extraction Before we can perform EL on the link candidates, we need to extract features from them that are characteristic for links to specific targets (see Sec. 5). All the feature values we compute for a single link candidate are called a *composite feature* when considered combined. Therefore, we retrieve one composite feature for each link candidate. As we have 20 million link annotations in Wikipedia, we also computed these features from those valid links as well as from invalid links so that we have comparable labeled data. Invalid links are links that we have generated and from which we know that they do not refer to their intended entity. This are, e.g., occurrences of organization aliases that do not signify any organization.

Entity linking Using the labeled composite features and the new ones extracted for each link candidate, we can reduce the task of EL to a classification problem: By training a classifier model with the ground truth data it can decide whether each link candidate links to an organization or not and if it does, to which. Ehmüller [3] describes how the classifier works in detail and which configuration leads to the best results.

Now we know where and which organizations German newspapers articles mentions. If a text refers to multiple businesses, it is likely that it describes a relation between them. Schneider [12] researched how we can extract those relations in order to enrich the *German Corporate Graph*.

HTML export The user of our generated *German Corporate Graph* may find a business relation that is unexpected, or simply want to comprehend how it was retrieved. In this case, it is necessary to provide a reference to the respective part of the original newspaper article, if applicable. An additional visualization of how we have extracted the relation makes this even more useful. Regarding the combined NER and EL process, we show where we have linked which tokens to which organizations. Therefore we enrich the original texts with HTML-links from the aliases to the Wikipedia article of their respective entity. We then export the HTML article to our Curation interface [6] with which the user can explore the results.

5 Features for Business Entity Linking

To perform EL, we have to describe an identified link candidate using appropriate features. On the one hand, they have to be expressive so that they allow us to identify the correct businesses, on the other hand, they have to be simple enough to be comparable for our classifier model. Inspired by the CohEEL project [5], we use three kinds of features:

1. The link score and entity score are **statistical** features.
2. The context score is a **linguistic** feature.
3. For the entity score and the context score, we additionally retrieve **second order features**.

Apart from the second order features, each of them is a real number in the interval $[0, 1]$. The second order features are provided in addition to the values of the entity score and context score and describe their proportion to "competing" values.

5.1 Link score

The most significant property of a link candidate is its alias a_{lc}. We use this to compute the link score ls, which denotes the probability that a_{lc} hyperlinks to a business inside the German Wikipedia:

$$ls(a_{lc}) = \frac{|\text{occurrences of } a_{lc} \text{ as link}|}{|\text{occurrences of } a_{lc}|}$$

This feature allows deciding whether a business alias signifies a business or not. For example, the alias "Bank" (English: "bank" or "bench") occurs 2200 times as a link but 44000 times for total. It may link to a certain organization, such as the German "Sparda-Bank" union. But in most cases, it denotes a bank in general or an ordinary bench, which is not a link in most cases. The small link score of 0.05 reflects that the alias "Bank" is not likely to denote a certain entity.

5.2 Entity score

As already mentioned, a business alias may signify different businesses in different contexts. For example "Telekom" occurs 700 times as a link. In 450 cases it refers to "Deutsche Telekom AG" and in 140 cases to "Telekom Deutschland GmbH", which is a subsidiary of "Deutsche Telekom AG".[8] As our system has to be aware of this ambiguity, we introduce the entity score es of an alias a_{lc} and an entity en as a second feature. Provided that a is a link to an entity, it is the probability that a_{lc} hyperlinks to en inside the German Wikipedia:

$$es_{en}(a_{lc}) = \frac{|\text{occurrences of } a_{lc} \text{ as link to } en|}{|\text{occurrences of } a_{lc} \text{ as a link}|}$$

For the alias "Telekom" this results in an entity score of 0.64 for the entity "Deutsche Telekom AG" and 0.20 for "Telekom Deutschland GmbH". These values allow our EL step to prefer those businesses that are more likely than others.

5.3 Context score

If our system would perform EL only based on the link score and entity score, it would never decide that an alias refers to an entity that has not the highest entity score for this alias. But depending on the context c_{lc} of a link candidate, it may be clear that the alias means an entity with a small entity score. As previously discussed, we can apply this to the sentence: "*Bosch liefert Servomotoren an die DB.*", which mentions "Deutsche Bahn AG" instead of "Deutsche Bank". Therefore we need a comparable mathematical representation of an link candidate's context.

[8] `https://www.telekom.com/de/konzern/weltweit/profile/die-deutsche-telekom-in-deutschland-336242`, last accessed on July 19, 2017.

5.3.1 Tf-idf contexts

We consider the bag of words of respectively 20 preceding and successive tokens (if existing) as sufficient for the disambiguation of a link candidate. This model counts how often which token occurred and disregards grammatical dependencies. We also perform stemming, which means that we normalize the tokens by discarding their grammatical forms. This generalizes the context even further and makes it easier comparable. Furthermore, we discard stop words that are very common in German texts and therefore considered as not expressive.

When we compare contexts, we want that those words have a stronger impact that are more significant. The more a word occurs in a bag of words and the less it occurs outside it, the more it is considered as significant. This is a well-known requirement in the field of information extraction and commonly heuristically solved using the tf-idf measure [11].

To calculate this, we have to define the following values: The *document frequency* df_t is the number of documents in the text corpus that contain t at least one time, which are Wikipedia articles in our case. For a total of N documents in the corpus, we calculate the *inverse document frequency* idf_t as:

$$idf_t = log\frac{N}{df_t}$$

The *term frequency* $tf_{t,b}$ is the frequency of a token t in a bag of words b. We finally can compute the *tf-idf-value* or *weight* $w_{t,b}$ for a token inside its bag of words:

$$w_{t,b} = tf_{t,b} \cdot idf_t$$

It describes the significance of the token. We use this measure to transform each bag of words b into a vector c_{lc} of tf-idf values for each contained token. We call this vector the *tf-idf context* of a link candidate. This numerical representation of a context is easily comparable to other tf-idf vectors.

5.3.2 Cosine similarity

To disambiguate an alias, we also compute a tf-idf vector c_{en} from each German Wikipedia article of an entity en. We assume that a link candidate is likely to refer to a specific entity, when its tf-idf context c_{lc} resembles the tf-idf vector of this entities' article, as they describe the same topic and probably use the same words. Thus, we compute the similarity between both vectors, which is our third feature, the context score cs. We use the cosine similarity, which is a standard approach to compare vectors [10, 11]:

$$cs_{en}(c_{lc}) = similarity(c_{lc}, c_{en}) = \frac{c_{lc} \cdot c_{en}}{|c_{lc}||c_{en}|}$$

This leads to a value in the interval $[0, 1]$. If it is 1, the contexts are completely similar; if it is 0, they have no token in common.

Thus, while the link score and entity score are statistical features that always have equal values for each pair of an alias and an entity candidate, the context score makes the system capable of disambiguating an alias based on its surrounding text.

5.4 Second order features

When the classifier disambiguates an alias, it has an only single link score, but one entity score and context score for each entity candidate. As not more than one entity may be correct, the entity scores and context scores are "competing" against each other. The classifier only makes a binary decision - whether an alias with a given context refers to a given entity or not - so it cannot consider how likely the alternative entities are.

Second order features are additional values for scores that represent their proportion to such competing scores. Using second order features, the classifier can regard, e.g., that an entity is likely for an alias, because it has the highest entity score, although it is relatively small.

For each entity score and context score, we add the following second order features:

1. The **rank** r is a natural number that denotes how many competing scores are greater than the respective scores (plus 1). It starts from 1 for the highest score(s).

2. The **absolute difference to the highest value** Δtop lies in the interval $(0, 1]$. The classifier should consider the highest score separately, what we achieve by using $+\infty$ instead of 0.

3. The **absolute difference to the next smaller value** $\Delta succ$ also lies in the interval $(0, 1]$. Since it is not defined for the smallest value, we use $+\infty$ for it.

In this way, we supplement each value for both the entity score and the context score with three additional values.

6 Preprocessing

To extract these features and use them to train a classifier for EL, we have to preprocess our knowledge bases.

6.1 Knowledge bases

Besides Wikipedia, we also use the knowledge base Wikidata, because it contains an ontology of Wikipedia's entities.

Wikipedia As we want to perform EL on German newspaper articles, the German Wikipedia provides appropriate training data. It currently consists of more than 3.6 million articles. Besides the texts, Wikipedia contains hyperlinks between the articles. They also may include infoboxes, which we discard. Their content is also part of DBpedia[9], which we already include in the German Company Graph.

For faster processing, the system accesses Wikipedia via a dump[10] that consists of XML and wikitext and has a size of 16 GiB.

Wikidata Wikidata[11] provides structured data for the entities of the German Wikipedia. It currently describes more than 24 million entities. This includes their classes, which can be, e.g., a business or, more generally, an organization. This information is important for us to identify which of Wikipedia's entities are relevant for the German Corporate Graph.

The system also accesses Wikidata via a dump[12]. It is stored in JSON format and has a size of 90 GiB.

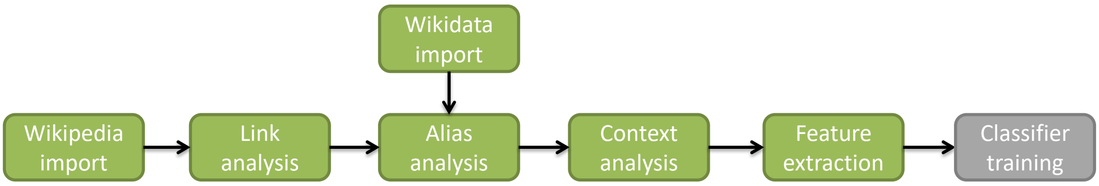


Figure 2: Steps of the preprocessing (green) and the final classifier training (gray)

6.2 Process

The preprocessing of the raw data consists of multiple steps as depicted in Figure 2. The following paragraphs describe them in detail.

Wikipedia import For each article in Wikipedia, we convert the wikitext to HTML using the Bliki library[13]. We then extract the raw text and all contained links to other Wikipedia entities.

Wikidata import To make use of Wikidata's ontology, we parse each of its articles and tag those entities that denote an organization. Similar entities in Wikidata and Wikipedia also have the same names so that we can apply this information to the corresponding Wikipedia entities.

Link analysis At first, the link analysis step refines the extracted Wikipedia links. This means that it removes links to missing pages and resolves links to redirect pages to their final target pages. Secondly, we count how often each alias of a link occurs as a link to calculate its link score. We also compute its entity score by counting how often each alias refers to which entity. By doing this also vice versa, we know how often an entity is referred by which aliases.

Alias analysis Most of Wikipedia's links are not relevant for us, as they do not refer to an organization. Using Wikidata's ontology, we could discard all those links. But instead, we use the statistic that we have generated during the link analysis, which lists all occurring aliases for a specific entity. By doing this, we find all aliases that may

[9] `http://wiki.dbpedia.org/`, last accessed on July 21, 2017.
[10] `https://dumps.wikimedia.org/dewiki/`, last accessed on February 6, 2017.
[11] `https://www.wikidata.org/`, last accessed on July 21, 2017.
[12] `https://dumps.wikimedia.org/wikidatawiki/entities/`, last accessed on February 6, 2017.
[13] `https://github.com/idio/bliki`, last accessed on July 21, 2017.

link to an organization. Only after we have determined these aliases, we remove all those links that have an alias, which never refers to an organization. These make up 90% of Wikipedia's links. In this way, the system retains links that do not link to any organization but have an alias of an organization. Those links are a valuable since their features provide negative training data for our classifier.

To gain even more training data for the classifier, we build a trie (a token based prefix tree) that contains all known aliases. Using this, we then find all occurrences of organization aliases in Wikipedia articles that do not refer to an entity. We assume that aliases, which also occur as a link in the same article, also mean the same entity. We store them as *extended links*, which serve as additional positive training data. If aliases have no corresponding link in the same article, we save them as *trie hits*, which serve as negative training data.

Context analysis To also compute the context features for the training data, we need to determine the tf-idf vectors for those links that we have collected. Therefore, we count the document frequency for each token in Wikipedia that is not a stop word. We also determine the term frequency of each token inside a link's or extended link's context and each entire Wikipedia article. This allows us to compute the tf-idf vectors for each link and entity, as described in Section 5.3.1.

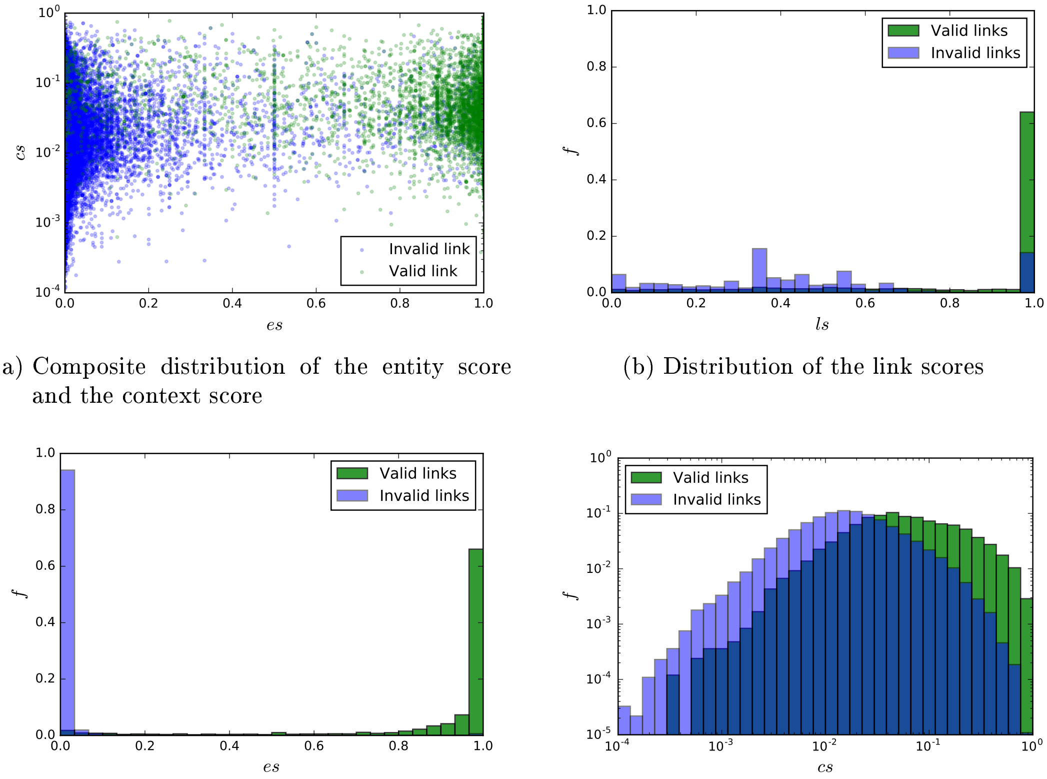
These extracted features for each of Wikipedia's relevant links constitute our training data. We use it for a classifier that performs business entity linking on newspaper articles.

7 Evaluation

Our system needs be suitable for economical use. We can express this as two requirements: On the one hand, the found combined NER and EL process should find as many correct references to organizations as possible. While Ehmüller [3] evaluates this in terms of the standard measures precision, recall and F_1 score, this thesis focuses on how significant the features are without considering a certain classifier model. On the other hand, our system has to be capable of processing large amounts of data in reasonable time. As our project aims at solving this using cluster computing, this evaluation considers how well the described preprocessing scales out.

7.1 Significance of the features

Given an entity en and a link candidate $lc = (a_{lc}, c_{lc})$ consisting of an alias a_{lc} and its context c_{lc}, the classifier decides whether lc references en or not. Note that, while the link candidate's entity score $es_{en}(a_{lc})$ and the context score $cs_{en}(c_{lc})$ depend on en, the link score $ls(a_{lc})$ only depends on the alias itself. Therefore these features must allow a distinction between valid and invalid links regarding the given entity. The following paragraphs evaluate the distribution of the features based on a sample of exactly 100,000 composite features from our training data and how expressive they are for this distinction.



(a) Composite distribution of the entity score and the context score

(b) Distribution of the link scores

(c) Distribution of the entity scores

(d) Distribution of the context scores

Figure 3: The link score, entity score and context score have significantly different distributions for valid and invalid links.

Figure 3a gives an overview of the composite distribution of the entity score and context score. Note that the context score is scaled logarithmically. The diagram reveals three properties:

1. There are two clearly identifiable **clusters**: One for valid links and a high entity score and one for invalid links and a small entity score.

2. The entity score is **more expressive** than the context score, as the clusters can only be separated using the entity score.

3. There is a **correlation** between the entity score and the context score: The upper right half of the diagram contains a lot more valid links than its lower left half. This means that the entity score and context score are "confirming" each other.

Now we consider the features separately to get a more concise understanding of their quality:

Link score Figure 3b shows the distribution of the link scores for the valid and invalid links. Here and in the following, the overlap between the bars is shaded in dark blue. As not more than one entity is valid for each link candidate, our sample contains only 8.4% valid links. Thus, for the better visibility, f shows the relative frequency of the link scores. It is 1 both for the valid and invalid links if added up. We can see that a valid link has a link score that is close to 1 in more than 60% of cases, while invalid links have a link score that is widely distributed. In fact, almost all of the links represented by the last bar have a link score, which is exactly 1. This means that there are a lot of explicit aliases, which always refer to the same entities.

Entity score Figure 3c shows the distribution of the entity scores. The extreme left and extreme right bar represent the clusters observed in Figure 3a. Hence the entity score is very expressive for our task. A classifier that decides based on a linear separation between these clusters should yield reasonable results. But since the of portion valid links is relatively small, this feature is not sufficient.

Context score Finally, Figure 3d shows the distribution of the context scores. Note that the frequency is scaled logarithmically. Although there is a large overlap between the valid and the invalid links, the distribution of the valid links is significantly shifted to greater context scores. For a context score of about 0.1, the context score has the least significance. The smaller or greater it becomes from there, the better the classifier can make its decisions.

Second order features Also the second order features have a high expressiveness, as shown in Figure 5 in the appendix. Similar to the three features described above, they show significantly different distributions for valid and invalid links. Thanks to their support, the classifier improves the results of the EL even further.

7.2 Scale out

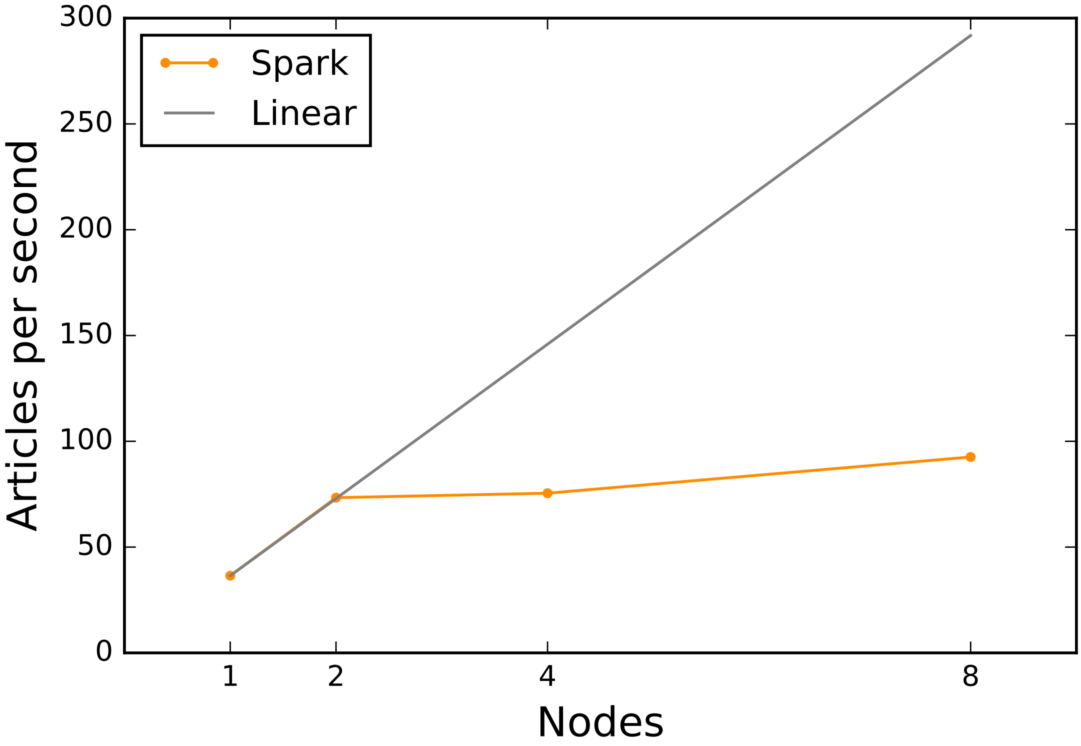


Figure 4: Scale out of the context score computation

To make our system scalable for arbitrary input sizes, we use the cluster computing framework Apache Spark[14]. Figure 4 shows the scale out of the context score computation for a sample of 100,000 Wikipedia articles. Starting from the bags of words, this comprises the calculation of the tf-idf vectors and its respective cosine similarities. The context score computation is one of our most time-consuming tasks.

We have calculated the efficiency as the average amount of articles that is processed within a second and measured the scale out for a different number of nodes (i.e., physical computers), while the number or CPU cores was constant. For the better interpretation

[14] `https://spark.apache.org/`, last accessed on July 20, 2017.

of the results, the diagram also depicts a linear scale out. It denotes that the efficiency increases in the same proportion as the number of nodes. Since there is always communication overhead between the nodes, which increases runtimes, a linear scale out or even super-linear scale out is difficult to achieve. As we can see, we reach a linear scale out for two nodes, but overall, the scale out is sub-linear. The smaller increase for more than two nodes might be explained by the relatively small input sample, which increases the proportion of the communication overhead. The other steps of our preprocessing show a similar scale out behavior.

To contextualize our approach, we compare the extraction of the bags of words and the subsequent tf-idf computation to another implementation for the same task. This implementation is part of the Spark MLlib[15], which was specially designed for cluster computing. 15 runs of both implementations for all links of the German Wikipedia led to an average time of 9.5 minutes for the Spark MLlib and 6.5 minutes for our approach. Therefore, it computes the contexts of link candidates 1.5 times faster than the Spark MLlib.

As shown above, the features we use for the extraction of business relations from text have a high quality. They allow a distinction between valid and invalid references to entities. Furthermore, our preprocessing is suitable for cluster computing. It can be distributed over multiple nodes and has an increasing efficiency for a higher amount of computing resources.

8 Conclusion and Outlook

Ambiguous business aliases make business entity linking in newspaper articles a complex problem. It becomes possible by the extraction of significant features, what we achieved by using statistical features, a linguistic feature and additional second order features. All of them turned out to have a significant correlation to whether an alias in a text refers to a certain business entity or not. Thus, we used them to perform combined NER and EL to determine where a newspaper article mentions which businesses. Thanks to this disambiguation of business aliases, we were able to analyze newspaper articles even further and extract business relations from them. We then inserted them into the German Corporate Graph that helps to get a better understanding of the German corporate landscape.

[15] `https://spark.apache.org/mllib/`, last accessed on July 20, 2017.

Regarding this full operative system, the next steps would be more focused on adding more data sources and testing more features.

The most promising improvement is to expand the training data. So far, we only used the German Wikipedia and Wikidata. But these do by far not cover all German businesses. Additional structured data sources, such as Implisense, would provide more business aliases. Those can be used to detect even more of their mentions in texts. Besides, we only used Spiegel Online to perform combined NER and EL. To extract more business relations, we should include other German newspapers with an economic focus, such as heise online[16].

Since many businesses do not have a Wikipedia article or only a few distinctive words in it, the enrichment of their tf-idf vectors would help to disambiguate aliases in a text. We can retrieve more of those words from, e.g, the homepage of the business or an article about its location. As some words are typical for an entire business sector, such as "Motor" for the automobile industry, this allows generating tf-idf vectors from multiple articles regarding the same sector. We can then use these vectors for businesses without an article.

It is also advisable to disambiguate aliases based on other decisions inside the same newspaper article. It is unlikely that the same alias references more than one business in one article, hence the EL process should prefer to link to those targets that were detected previously in it.

By implementing these improvements, the system may become capable of determining where a text refers to which businesses for most of the German companies. This would eventually help to make a lot of the knowledge of enterprise related German newspaper articles machine-understandable.

Acknowledgments

This project would not have taken place without the help and confidence of certain persons, who I would like to acknowledge.

First of all, I thank Felix Naumann, who initiated this project, for his interest and helpful advice. He took himself a lot of time to talk with us about our progress and drew attention to the priorities, when necessary.

An extraordinary word of thanks goes to my supervisors, Toni Grütze and Michael Loster,

[16] `https://www.heise.de/`, last accessed on July 20, 2017.

for their devoted guidance and assistance during the project and in improving this thesis. They were always available when there were open questions.

I would also like to acknowledge our business partners from the Commerzbank AG, Oliver Maspfuhl and Dirk Thomas, for supporting us with the necessary hardware and additional data. They have pointed out why the German Corporate Graph is such a meaningful instrument and its creation an important objective.

References

[1] Automobilwoche. Abgas-Skandal: VW-Zulieferer kämpfen mit Kurzarbeit. `http://www.automobilwoche.de/article/20161018/AGENTURMELDUNGEN/310189944/abgas-skandal-vw-zulieferer-kaempfen-mit-kurzarbeit`, 2016. (last accessed on July 21, 2017).

[2] Statisches Bundesamt der Bundesrepublik Deutschland. Unternehmensregister. `https://www.destatis.de/DE/ZahlenFakten/GesamtwirtschaftUmwelt/UnternehmenHandwerk/Unternehmensregister/Tabellen/UnternehmenBeschaeftigteUmsatzWZ08.html`, 2016. (last accessed on July 21, 2017).

[3] Jan Ehmüller. Evaluation of Entity Linking Models on Business Data, 2017.

[4] Imelda Flaig. VW-Abgasaffäre belastet Zulieferer im Land. `http://www.stuttgarter-zeitung.de/inhalt.baden-wuerttemberg-vw-abgasaffaere-belastet-zulieferer-im-land.06ea4637-d060-4109-ad3c-c341a04bf1bd.html`, 2016. (last accessed on July 21, 2017).

[5] Toni Gruetze, Gjergji Kasneci, Zhe Zuo, and Felix Naumann. CohEEL: Coherent and efficient named entity linking through random walks. *Web Semantics: Science, Services and Agents on the World Wide Web*, 37:75–89, 2016.

[6] Milan Gruner. Analysis and Simplification of Business Graphs, 2017.

[7] Alexander Immer. Alias Generation to Improve Company Recognition in Text, 2016.

[8] Lando E. N. Löper and Matthias Radscheit. Evaluation of Duplicate Detection in the Domain of German Businesses, 2017.

[9] Leonard Pabst. Efficient Blocking Strategies on Business Data., 2017.

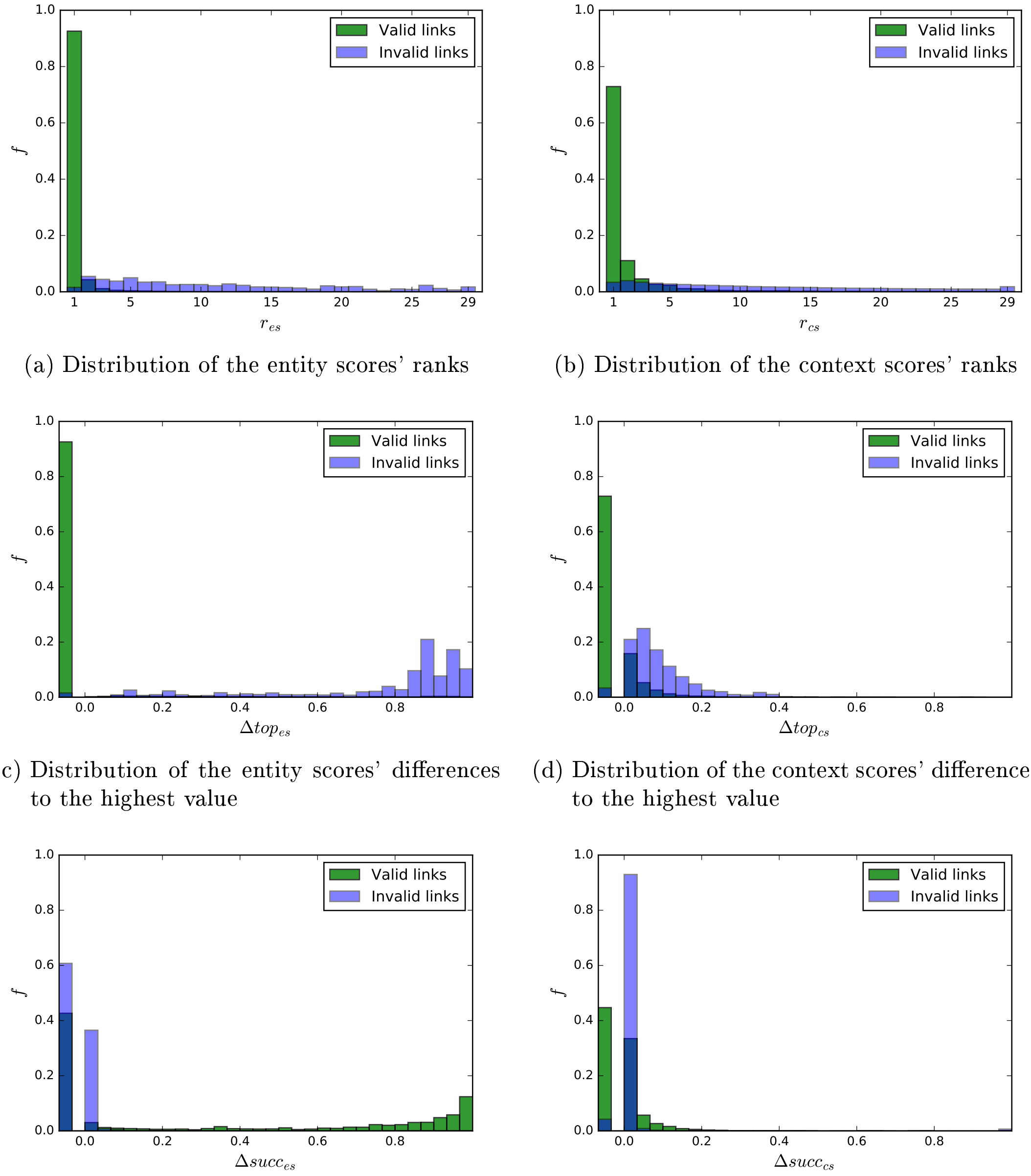
[10] Baeza-Yates Ricardo et al. *Modern information retrieval.* Pearson Education India, 1999.

[11] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

[12] Alec Schneider. Evaluation of Business Relation Extraction Methods from Text, 2017.

[13] Nils Strelow. Distributed Business Relations in Apache Cassandra, 2017.

Appendix



(a) Distribution of the entity scores' ranks

(b) Distribution of the context scores' ranks

(c) Distribution of the entity scores' differences to the highest value

(d) Distribution of the context scores' differences to the highest value

(e) Distribution of the entity scores' differences to the next smaller value

(f) Distribution of the context scores' differences to the next smaller value

Figure 5: The second order features have a significantly different distribution for valid and invalid links. For (c), (d), (e) and (f), the bin at the very left represents the value $+\infty$.

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used any other than the declared resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Potsdam, July 21, 2017

. . . .
Jonathan Janetzki