ORIGINAL ARTICLE

Extracting supply chain maps from news articles using deep neural networks

Pascal Wichmann[a], Alexandra Brintrup[a], Simon Baker[b], Philip Woodall[a], and Duncan McFarlane[a]

[a]Institute for Manufacturing, University of Cambridge, United Kingdom; [b]Language Technology Lab, University of Cambridge, United Kingdom

ARTICLE HISTORY
Compiled September 7, 2019

ABSTRACT
Supply chains are increasingly global, complex and multi-tiered. Consequently, companies often struggle to maintain complete visibility of their supply network. This poses a problem as visibility of the network structure is required for tasks like effectively managing supply chain risk. In this paper, we discuss automated supply chain mapping as a means of maintaining structural visibility of a company's supply chain, and we use Deep Learning to automatically extract buyer-supplier relations from natural language text. Early results show that supply chain mapping solutions using Natural Language Processing and Deep Learning could enable companies to a) automatically generate rudimentary supply chain maps, b) verify existing supply chain maps, or c) augment existing maps with additional supplier information.

KEYWORDS
supply chain management; supply chain map; natural language processing; text mining; supply chain visibility; supply chain mining; deep learning; machine learning

1. Introduction

Complex products, like cars or aircraft, can be composed of tens of thousands or even millions of parts. Rather than all being produced in-house, parts and materials are sourced from a large number of suppliers spread across the world. As substantial parts of the value creation are outsourced to suppliers, who, in turn also outsource to sub-tier suppliers themselves, increasingly multi-tiered, complex and geographically distributed supply networks emerge (Christopher and Lee 2004). Consequently, companies gradually lose visibility over the topology of their supply network. A study by Achilles, a provider of supply chain management solutions, claims that "40% of companies who sourced only in the UK, and almost 20% who sourced globally, had no supply chain information beyond their direct suppliers" (Achilles Group 2013).

Lacking visibility of the supply chain structure poses a problem: By definition, a supply network is a network of dependencies to suppliers, and the performance of a company's supply chain is crucial to its operations. Information about its extended suppliers is a valuable input to various decision-making processes of a firm, such as

CONTACT P. Wichmann. Email: pw351@cam.ac.uk

managing the efficiency, resilience, or sustainability of its supply chain. Furthermore, in recent years companies have come under increasing pressure to understand their supply chains to prevent modern forms of slavery and other human rights violations. In particular, supply chain risk management without visibility of the supply network is a problem while at the same time the emergence of longer, geographically distributed supply chains exposes companies to more and a wider range of risks. Disruptions that occur on a sub-tier can propagate through the network, creating a ripple effect (Ivanov, Sokolov, and Dolgui 2014) and halt the production of companies that never knew they had this dependency on a sub-tier supplier. Studies show that the share of supply chain disruptions that originate with suppliers further upstream than the direct suppliers can be as high as 50% (KPMG International & The Economist Intelligence Unit 2013; Business Continuity Institute 2014). Suppliers critical to continued operations can be located anywhere in the multi-tiered network and do not have to correspond to large sales volumes (Yan et al. 2015). The reason why structural supply chain visibility cannot easily be achieved is a combination of multiple factors. One major reason is that companies consider information about their supply base proprietary and are unwilling to share it. Supply chain mapping is frequently named as the recommended solution to the problem of limited supply chain visibility, and various tools exist for visualising buyer-supplier relations, yet the actual issue of acquiring the required data in the first place remains unaddressed (Farris 2010).

Even though data that can readily be used for supply chain mapping is still scarce, vast amounts of data have become abundantly available at low cost via the Web. A large proportion of this data is in text form and contains valuable information about buyer-supplier relations. Since these text documents typically consist of running text in natural language (unstructured text) instead of in tables with a known data schema, extracting information from it is a challenging problem. Thanks to advances in Machine Learning (ML), in particular Deep Learning, and Natural Language Processing (NLP), the extraction of information can now be increasingly automated. A solution that could at least partially automate the generation of supply chain maps from text documents would have many beneficiaries and use cases. Improved knowledge of the supply chain structure would enable a company to better detect and mitigate risks in advance. For example, a company may not be aware that some of its direct suppliers depend on the same sub-tier supplier, a potential single point of failure. In this case, the true risk exposure is obscured. With this knowledge, a company could mitigate the risk, e.g. by increasing inventory levels, demanding suppliers to diversify their supply base, or by identifying substitute parts with different supply chains. Knowledge of the supply chain structure would also enable a company to react to risk events more quickly and appropriately. For example, knowledge about which sub-tier suppliers are located in a recently flooded region could allow a company to react quickly and, for example, enable them to secure alternative supplies faster than any competitor. The potential benefits are not limited to companies managing risks of their own supply chains. Other actors may also benefit from a better understanding of supply networks, such as governmental agencies, insurance companies, or management consultancies.

The overall aim of this research is to examine to what extent and how supply chain maps can be automatically generated from unstructured, natural language text, such as news reports or blog posts. Given how frequently new algorithms and network architectures are proposed in ML and NLP, the aim is *not* to identify the best possible algorithm but to test the general feasibility of the idea. A prerequisite for the creation of supply chain maps is the extraction of individual buyer-supplier relations which represents the main focus of this paper. Figure 1 shall summarise the problem background.

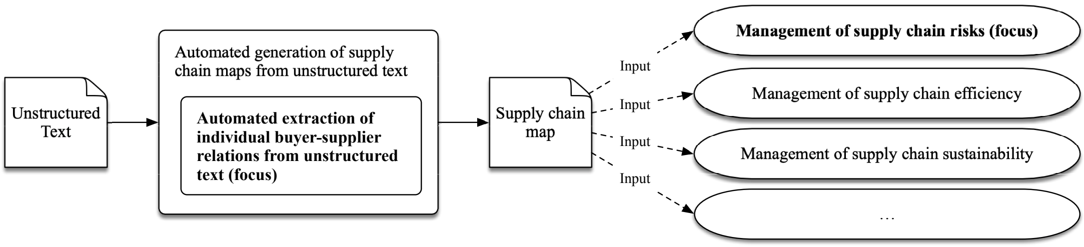

Figure 1. Conceptual model of the research focus: Automating the extraction of individual buyer-supplier relations from unstructured text

This study builds upon a previous paper (Wichmann et al. 2018) where the idea of automatically generating supply chain maps from natural language text was first introduced and its challenges discussed. In this paper, we focus on the automated classification of buyer-supplier relations by creating a text corpus and using it to train and test a Deep Learning classifier. The automatically extracted buyer-supplier relations are then visualised in a basic supply chain map.

After summarising the relevant background (Section 2), namely supply chains, supply chain visibility as well as relevant concepts from Machine Learning and NLP, we define the problem of extracting individual buyer-supplier relations from text (Section 3). We then outline the methodology for addressing the problem (Section 4). Subsequently, we summarise and discuss the results (Section 5). The extracted buyer-supplier relations can be visualised in form of a basic supply chain map (Section 6). Finally, we provide concluding remarks and propose ideas for future research (Section 7).

2. Related work

2.1. *Supply chains and supply chain mapping*

2.1.1. Supply chains

A supply chain emerges as a focal company (hereafter also referred to as Original Equipment Manufacturer or OEM) buys products or services from a supplier to produce their own products. Since supply chains are networks (Lambert and Cooper 2000), they consist of nodes and directed links of “flows of products, services, finances, and/or information from a source to a customer” (Mentzer et al. 2001). The combination of nodes and links give the network its structural dimensions. The horizontal structure refers to the number of tiers across the supply chain. The vertical structure refers to the number of suppliers or customers represented within each tier (Lambert and Cooper 2000). The term “upstream” is used to denote the direction towards to original supplier whereas “downstream” refers to the direction towards the ultimate customer.

2.1.2. Structural supply chain visibility

In academic literature, the term *supply chain visibility* (also referred to as *supply chain transparency*) has been defined in various ways. For a comprehensive overview the reader may refer to Goh et al. (2009). Within the scope of this paper, we adopt the broader definition by Barratt and Oke (2007) who define supply chain visibility as "the extent to which actors within the supply chain have access to or share timely information about supply chain operations, other actors and management which they consider as being key or useful to their operations".

Included in the above definition is knowledge of the topology of the supply network, that is knowledge of the actors and the network of their dependencies, which we will refer to as *structural* supply chain visibility. Structural supply chain visibility is often limited (e.g. (Achilles Group 2013)). Any company knows its direct suppliers and customers, yet already knowledge of second-tier suppliers tends to be incomplete.

The reason for limited supply chain visibility is a combination of multiple factors. The main reason is the "proprietary nature of each supplier's relationships with its partners" (Sheffi 2005). Suppliers have an incentive not to disclose their own supply network to their customers, especially if they run the risk of being cut out as the middleman or losing bargaining power. Suppliers can be contractually obliged by an OEM to disclose their own suppliers, but the information asymmetry between OEM and direct suppliers renders it difficult for the OEM to check the completeness of the provided information. The difficulty of obtaining the required data is exacerbated by the fact that supply chains are dynamic (Lambert and Cooper 2000).

Lacking *structural* visibility cannot be simply addressed by track-and-trace technology based on RFID or other IoT solutions. If used across company boundaries, participating companies know each other and have consented to exchange real-time information about the location and condition of goods in transit, inventory levels or other dynamic aspects of supply chain performance. However, these technologies have not been designed to discover the supply chain structure, such as otherwise unknown supply chain participants on a sub-tier and their inter-relations.

2.1.3. Importance of structural supply chain visibility

The importance of structural supply chain visibility has been highlighted by various studies: Basole and Bellamy (2014) examine the link between structural supply chain visibility and risk management and find that "structural visibility into the lower tiers of the supply network has a significant mitigating impact on cascading risks" and that "enhanced visibility is an important and perhaps essential capability for effective supply chain risk identification and mitigation. Supply chain managers must therefore move beyond a simplified dyadic or triadic view to a more holistic approach when developing risk identification and mitigation strategies". Examples of obscured risk include suppliers depending on the same sub-tier supplier or high-risk supply chain participants on a sub-tier. Yan et al. (2015) introduce the idea of a "nexus supplier". Contrary to the intuition that strategic, direct suppliers are the critical ones due to their direct and large impact on a buying firm's profit and risk position, a nexus supplier could be located in any (sub-)tier of the supply chain, does not have to relate to a large sales volume, but has a potentially large impact on the buying firm if it was disrupted. The existence and identity of such a nexus supplier on a sub-tier could only be revealed with better visibility into the supply chain structure. The network structure also determines how risk events propagate through the network and if they get absorbed or even amplified (Jüttner, Peck, and Christopher 2003). An early detection of and response to risk events would require knowledge about which events are relevant to a company's supply chain. For this, too, knowledge of the supply chain structure is necessary. Christopher and Peck (2004) state that a "fundamental pre-requisite for improved supply chain resilience is an understanding of the network that connects the business to its suppliers and their suppliers and to its downstream customers. Mapping tools can help in the identification of 'pinch points' and 'critical paths' ".

2.1.4. Supply chain mapping

Supply chain maps are "a representation of the linkages and members of a supply chain along with some information about the overall nature of the entire map" (Gardner and Cooper 2003) and aim to address the problem of limited structural supply chain visibility. The purpose of supply chain maps, and hence the scope and level of detail, can vary (Gardner and Cooper 2003). Their purpose is generally strategic and they range from a geographic vulnerability map which "simply depicts which supplier of what parts are located in each area of the world" (Sheffi 2005) to maps that show "the flow of parts out of given regions, depicting who is involved and the plants in other parts of the world that are dependent on them" (Sheffi 2005). Supply chains may or may not depict actual geographical relationships (Gardner and Cooper 2003). Gardner and Cooper (2003) provide a comprehensive overview of examples of supply chain maps. The minimal set of elements of these supply chain maps typically consists of the companies (nodes of the network) and their inter-relations (arcs of the network). The arcs commonly indicate the flow of goods but may also indicate flows of information or money. In this paper, we refer to *supply chain mapping* as the *overall process* of creating and maintaining a supply chain map. This process includes the steps of gathering the information needs, acquiring and analysing the information and visualising the results on the required aggregation level. Only few papers appear to exist that reflect on supply chain mapping as a method, e.g. Gardner and Cooper (2003) and Farris (2010). Other papers approach the topic from the perspective of lean management by extending value stream mapping to supply chains, e.g. Suarez-Barraza, Miguel-Davila, and Vasquez-García (2016). However, numerous papers report on the *application* of supply chain mapping to specific scenarios. E.g. Choi and Hong (2002) provide supply chain mapping case studies in the automotive industry. The supply chain maps were limited to the centre console assembly of three different product lines. The data was collected *manually* through interviews, from documents provided by the automotive companies, and via observations during a plant tour. Choi and Hong (2002) compare the three resulting supply network structures from the points of view of formalisation, centralisation, and complexity. Another example of a manual supply chain mapping exercise is a report by the US Geological Survey. This report "uses the supply chain of tantalum (Ta) to investigate the complexity of mineral and metal supply chains in general and show how they can be mapped" (Soto-Viruet et al. 2013).

2.2. *Deep Learning and Natural Language Processing*

2.2.1. Supervised Learning

Today, one can arguably distinguish three main types of machine learning: supervised learning, unsupervised learning, and reinforcement learning (Murphy 2012). In *supervised learning*, the objective during a training phase is to learn a mapping from provided inputs (called features) to provided outputs (called labels). After the training phase, this learned mapping can be applied to predict the labels for previously unseen inputs, as shown in Figure 2. If the label is a discrete value, the problem is called *classification*, and *regression* otherwise.

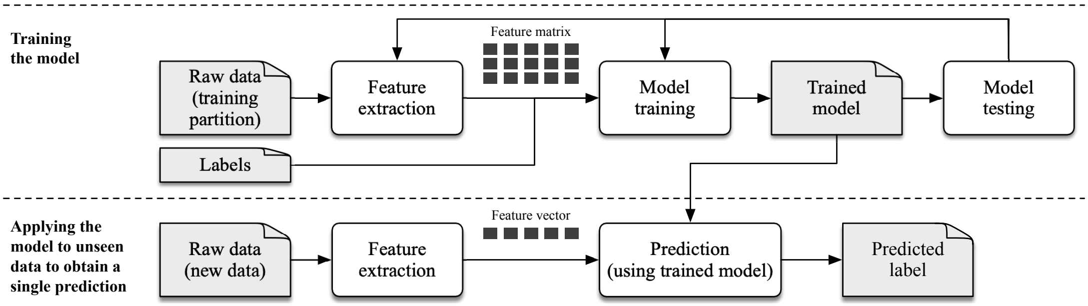

Figure 2. Illustration of a supervised learning process

2.2.2. Neural Networks and Deep Learning

Neural Networks (NN), or Artificial Neural Networks (ANN), are a family of machine learning algorithms that are loosely inspired by the biological brain. Each unit in a neural network "resembles a neuron in the sense that it receives input from many other units and computes its own activation value" (Goodfellow, Bengio, and Courville 2016). Neural networks are composed of stacked layers, and those layers between the input layer and the output layer are called *hidden* layers. The number of layers determines the *depth* of the model, hence the name *"Deep Learning"* for neural networks with multiple hidden layers. Comprehensive introductions to Deep Learning can be found in Chollet (2018) and Goodfellow, Bengio, and Courville (2016).

A wide variety of model architectures have been developed in recent years. *Feed-Forward Neural Networks* are the simplest form of neural networks. The name refers to the fact that the connections between the nodes in the network do not form a cycle. A *multi-layer perceptron* (MLP) is a type of feed-forward neural network with input layer, output layer and at least one hidden layer. *Recurrent neural networks* (RNN), as opposed to feed-forward neural network, contain loops. This way, they allow the behaviour of neurons not just to be determined by activations in previous hidden layers but also by activations at earlier times or even a neuron's own activation at an earlier time. RNNs are particularly suited to sequential data, such as text sequences, since they can consider the order of the sequence for a prediction task. *Long short-term memory* (LSTM) networks are a type of RNN that contains LSTM units. LSTM units were introduced by Hochreiter and Schmidhuber (1997) and address the problem of the vanishing gradient. *Bidirectional* RNNs combine the different representations learned from reading data in both directions. For some types of sequential inputs, models perform similarly well if the data is read "anti-chronologically". However, because these RNNs trained on the reversed sequence learn a different representation, it is useful to combine the outputs of RNNs trained on the normal *and* the reversed sequence (Chollet 2018). Such network architectures are called *bidirectional* RNNs, and bidirectional versions also exist for RNN sub-types (e.g. *BiLSTM* (Graves and Schmidhuber 2005)). Even more recent developments, such as Google's attention-based transformers (Vaswani et al. 2017), were not considered within the scope of this paper.

2.2.3. Natural Language Processing

Natural Language Processing (NLP) is a rapidly developing sub-field of Artificial Intelligence (AI) that specialises in the extraction and manipulation of natural language text or speech (Chowdhury 2003). Modern NLP methods increasingly rely on Machine Learning, in particular Deep Learning. In this work, we focus on Information Extraction (IE), a fundamental task of NLP that aims to automatically extract structured information from unstructured natural text (Cowie and Lehnert 1996). This structured information is typically used to construct large knowledge bases, relational databases, and ontologies. IE is subdivided into two subtasks: *Named Entity Recognition (NER)*, which is the subtask of locating and classifying instances (text mentions) of entities with pre-defined categories of interest (Nadeau and Sekine 2007), and *Relation Extraction (RE)*, which is the task of detecting and classifying semantic relationships between named entity mentions (Bach and Badaskar 2007).

In analogy to Machine Learning in general, relation extraction methods can be distinguished based on the degree of supervision. They commonly fall into one of the following categories (cf. Mintz et al. (2009)): *Unsupervised methods* simply use statistical co-occurrence, *supervised methods* require hand-labelled examples to learn from, *distant supervision* attempts to address the costs of obtaining labels by leveraging a database of known relations, and *bootstrapping* is an iterative process starting with a few seeds but suffering from semantic drift. Lastly, *lexico-syntactic patterns*, such as the Hearst patterns (Hearst 1992), are manually pre-defined and do not use Machine Learning.

Machine Learning algorithms generally expect numeric tensors as input. In order to use a sequence of text as input, it first needs to be broken down into tokens (tokenisation), and then each token needs to be converted into a numeric vector (vectorisation). *Word embeddings* are real-valued, low-dimensional, and dense vectors that represent unique words (Mikolov et al. 2013) and encapsulate semantic relationships between different words. Word embeddings that have been pre-trained on large datasets are available, such as GloVe[1] or Google's Word2Vec News embeddings[2].

Generally used performance metrics for information retrieval and information extraction systems include *precision*, the share of retrieved documents that are relevant, and *recall*, the share of all relevant documents that are retrieved. Because of the trade-off between both metrics, the harmonic mean of both – the F_1 score – is commonly used for benchmarking.

2.3. *Automatic extraction of supply networks from text*

Farris (2010) attempts to address the problem of finding actual data for use in strategic supply chain mapping by using economic input-output data. This data was then converted into macro industry supply chain maps. However, the process was manual and did not allow for any maps on a company-level.

NLP can be used to automatically generate general network structures from text, e.g. to automatically extract taxonomic and non-taxonomic ontologies from text (Maedche and Staab 2001). Ontologies form a network structure of directed relations which can be visualised in an ontology graph. The scope of related work, such as in the field of OpenIE, is commonly still limited to basic relations, such as "is-a"

[1]Global Vectors for Word Representation (GloVe); `https://nlp.stanford.edu/projects/glove/`; last accessed: 2018-01-07

[2]Word2Vec; `https://code.google.com/archive/p/word2vec/`; last accessed 2018-01-07

or "located-in" relations.

Extracting network structures from text has also been tried in the bio-medical domain. The LION project[3] (Pyysalo et al. 2018) uses statistical co-occurrence to automatically extract relations from scientific papers in the bio-medical domain and visualise them as an interactive network. The purpose of the proposed method and tool is to facilitate the discovery of new knowledge. Because the system uses co-occurrence only, relations are non-directional and not further classified.

A recent paper by Yamamoto et al. (2017) attempts to extract company-to-company relations from text but only focusses on (non-)cooperative and (non-)competitive relations using distant supervision and manual labels. While this classification may seem to also be helpful with the extraction buyer-supplier relationships, it actually is not: Competitive clues include terms like "sues", "lawsuit" or "loses". Because buyers and suppliers not uncommonly happen to have legal disputes, these clues can be misleading for the purpose of supply chain mapping.

In a previous paper, we introduced the idea of using NLP to automate the supply chain mapping process, derived a set of requirements for such a process and showed a basic prototypical implementation of a system (Wichmann et al. 2018). The relation extraction was based on lexico-syntactic patterns and, therefore, showed a high precision but suffered from low recall. To capture a wider range of expressions and reduce the effort of manually defining extraction patterns, we proposed to use Deep Learning.

3. The problem of extracting buyer-supplier relations

3.1. *From the overall problem to the classification of individual relations*

An ideal solution for automating the complete process of supply chain mapping from text has to meet a wide range of requirements, such as the ability to infer actual subtier relations ("transitivity problem") (Wichmann et al. 2018). However, the extraction of *individual buyer-supplier relations between two companies* can be considered a fundamental building block for automating the overall process. A collection of extracted individual relations could already be directly visualised as a (non-transitive) network. Conceptually, the extraction process requires two stages: First, mentions of named entities need to be detected and classified as organisations (as opposed to locations, persons etc.). Secondly, each pair of two organisational mentions needs to be classified with respect to the stated relationship between them. While general solutions for Named Entity Recognition (NER) are available, models to classify buyer-supplier relations do not appear to exist.

Figure 3 illustrates the focus of this paper conceptually.

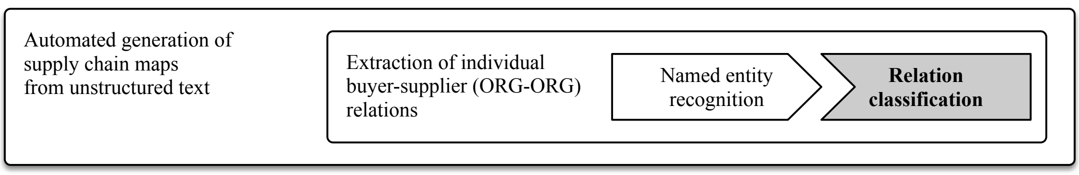

Figure 3. Focus of this paper: classification of buyer-supplier relations between two mentions of organisations

[3]LION project; `http://lbd.lionproject.net`; last accessed 2018-01-07

3.2. *Problem formalisation*

For a given single, self-contained sentence in the English language, all pairs of detected organisational entity mentions shall be classified with respect to the existence of a buyer-supplier relation explicitly stated between them. We will ignore relations that would require the resolution of pronouns to company names (co-reference resolution). Furthermore, we will ignore more multi-faceted relations at this stage, such as extracting supplied products or further companies involved. Each relation shall only be assigned one single class, so that the overall problem can be characterised as a *single-label multi-class classification problem* with classes, such as “company A supplies company B” , “company A is supplied by company B” (inverted direction), “company A and B engage in a partnership” or “no buyer-supplier relation expressed”. The class definitions used can be found in the methodology section below.

Across a collection of documents, the problem can be illustrated as shown in Figure 4: A collection of documents is converted into a list of triples. Each triple consists of two organisational named entities and the identified relationship class.

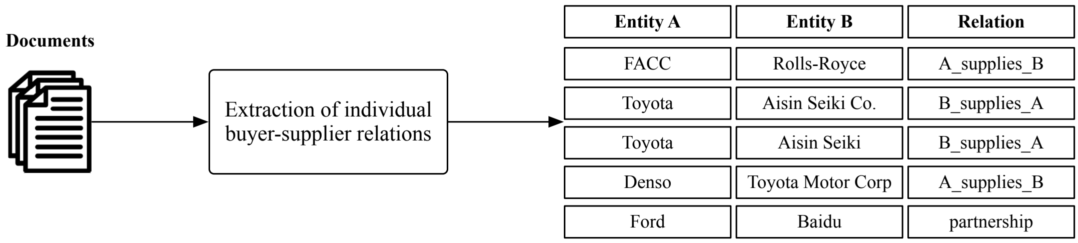

Figure 4. Buyer-supplier extraction across multiple documents: The aim is to extract triples of Entity A, Entity B, and the relationship class.

4. Methodology

4.1. *Overview*

We address the problem in two subsequent stages: corpus creation and relation classification.

Corpus creation The corpus creation consists of both the collection and preparation of the text input as well as the process of labelling sentences by human annotators. The importance of the corpus is two-fold:

- *“Gold standard”:* In order to evaluate classification performance, a human-labelled text corpus is required that will be considered the ground truth against which the classifiers’ predictions can be compared. The gold standard data will allow us to measure both recall and precision. The datasets also serves as a training dataset for a Machine Learning classifier and is, thus, more than a pre-processing step but part of the solution.
- *Suitability for automation:* Furthermore, higher inter-annotator agreement suggests a more manageable, formalisable task that is more likely to be suitable for automation. If it is impossible to establish a ground truth among human annotators, a classifier cannot be expected to perform well on the problem. It is not obvious that the task of classifying buyer-supplier relations is simple or

formalisable enough for annotators to agree.

The text corpus shall be representative of a general news set, such that a classifier's performance measured on the dataset is a good predictor of its performance on a previously unseen general news dataset. This is a challenging requirement given the expected small size of the text corpus and the low share of sentences describing buyer-supplier relation in a general news dataset.

Relation classification Only once a gold standard dataset has been established, a supervised classifier can be trained and the best-performing one with respect to recall, precision and F_1 score can be identified. The achievable performance is dependent on the size and quality of the corpus. A high recall would *not* yet suggest that supply chains can be fully reconstructed; this would depend on the information availability which is not tested in this stage. Similarly, NER errors are not considered when the classification performance is measured.

4.2. *Corpus creation*

4.2.1. Sentence collection

The aim of this phase is to create a pool of sentences from which sentences can be randomly selected and presented to the annotators.

To draw from a wider range of general news articles, we selected multiple data sources: the Reuters corpora TRC2 and RCV1[4], the NewsIR16 dataset[5], and a customised dataset obtained from webhose[6]. For an unbiased dataset, sentences should ideally be randomly sampled from these general news datasets. However, limited labelling resources are a constraint. Because sentences need to be manually annotated, the dataset cannot be too sparse so that annotators spend most of their time annotating sentences without any buyer-supplier relation (henceforth referred to as negative examples, whereas positive sentence express at least one directed or undirected buyer-supplier relation or partnership). Because most sentences in any news dataset are negative, annotated negative sentences are not that valuable.

Approach A: Sampling of documents into 3 partitions Our first approach to address this trade-off was to sample documents into 3 separate partitions: one partition for random documents drawn from a general news dataset, a second partition for documents that were retrieved using keywords related to selected key industries (aerospace and automotive), and a third partition for documents that were retrieved based on a search for company names in these key industries. This way, the trade-off between the expected relevance of a sentence and its bias could in principle be steered by adjusting the proportion of each partition in the final sample.

Among other reasons, aerospace and automotive were chosen as key industries as they are known for having complex and global supply chains. In addition, the assumption was that these industries are both well-covered in general news as well as that news reports often include supply chain information. For the aerospace industry, for example, documents were filtered for the existence of keywords, such as

[4] Reuters corpora TRC2 and RCV1 (`https://trec.nist.gov/data/reuters/reuters.html`)
[5] NewsIR16 dataset (`https://research.signal-ai.com/newsir16/signal-dataset.html`); last accessed: 2019-01-10
[6] Webhose.io (`https://webhose.io/`); last accessed: 2019-01-10

"aerospace", "aircraft" or "planemaker". The RCV1 Reuters dataset contained industry codes ("Thomson Reuters Business Classification") and could more accurately be filtered using 23 of these codes instead. 50% of documents were sampled from the aerospace industry and 50% from the automotive industry. To filter documents by company names, a of the top 100 global automotive company names and brands was used as well as a list of the top 100 global aerospace and defence companies. Relevant documents in each partition were subsequently segmented into individual sentences that would then be drawn randomly. Initial tests quickly revealed that the proportion of positive sentences even in the more relevant data partitions was too small for an efficient annotation by humans.

Approach B: Manual collection of candidate sentences To address this problem but without compromising the overall human annotation, in addition to the already created dataset, candidates for positive sentences were manually collected by 3 researchers and stored in a further data partition. To prevent biases, these positive sentences could not just be obtained via a Web search that used potential features, such as "supplies Toyota with". Otherwise, a classifier trained on the data would be biased towards the patterns the positive sentences were found with in the first place. Instead, the following strategies were deemed acceptable and used to collect the sentences:

- Using a Web search engine by using as a search term (a) a single company name or (b) the names of two companies of which one is known or merely suspected to supply the other.
- Manually analyse websites that tend to publish industry news, such as recent deals and partnerships.

In all of these cases, sentences were manually identified in the search result summaries, headings or the original articles that could describe a buyer-supplier relation, partnership or collaboration. Ambiguous sentences were not ignored but were also collected so that the overall dataset was rather too inclusive than too exclusive. Similar to the previous approach, the focus was on aerospace and automotive companies but any by-catch from other industries was also added to the collection. These candidate positive sentences were collected and stored without a label as it was up to the annotators to classify the sentences.

The drawback of adding manually collected candidate positive sentences is the introduction of additional bias. This is unavoidable as it is a direct consequence of the objective of manually collecting sentences. But it may lead to so-called "overfitting" and result in false positives if the classifier is applied to previously unseen data. Some words may be reliable indicators of buyer-supplier relations in the training data but not as reliable in a random general news dataset. E.g. words, such as "award" or "buy" may be *over-represented* in the *positive* examples of the training data. To address the issue, the classifier can be reiteratively improved: by manually labelling sentences that turned out to be false positives and adding these labelled sentences to the training data.

Overall sampling process The overall sampling and pre-processing methodology is shown by Figure 5 and combined both sampling approaches to equal parts. Documents were segmented into sentences using spaCy[7] as off-the-shelf solution. To facilitate

[7] spaCy (`https://spacy.io/`; last accessed: 2019-01-09) is a widely used open-source NLP software library

the subsequent annotation, sentences were automatically NER-tagged, again using spaCy. The organisation names were *not* provided to the NER tagger in advance. To reduce false positives, Flair[8] and the Stanford CoreNLP[9] NER taggers were used in combination with spaCy in a simple ensemble. The results of all three libraries had to match for an organisational named entity to be considered in the subsequent steps. Only sentences with two or more detected organisational named entities were admitted to the annotation process. Automatic NER tagging performs well but is still imperfect and organisations may not have been detected, erroneously detected, or detected with incorrect segmentation boundaries.

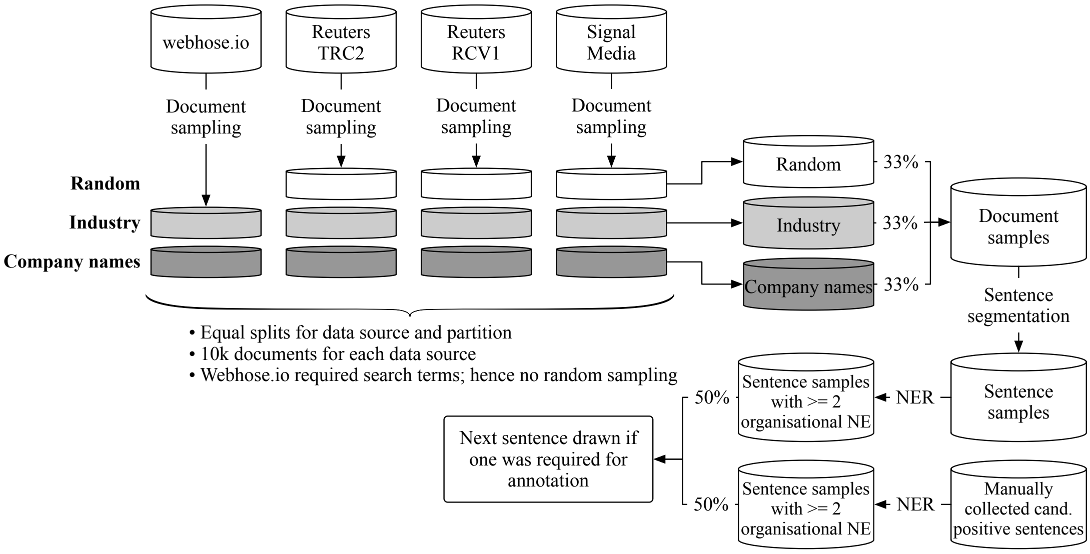

Figure 5. Documents were sampled from four different sources and assigned to 3 different partitions each. Manually collected candidate positive sentences were added to the corpus to increase the share of positive examples.

4.2.2. Labelling

Classes Due to the importance of directionality in supply relations, two classes are designed for explicitly expressed directed buyer-supplier relations: (a) "company A supplies company B", and (b) "company A is supplied by company B" (inverted direction). The actual sentence in the news article may use different words to express this relation, such as "purchasing from" or "using parts from". These relations are not limited to the purchase of goods, parts and material but also include the use of services, such as logistics services. These relations may be expressed in any tense (past, present or future) since the tense could later automatically be identified if necessary. Furthermore, these relations should be expressed as certain, factual statements rather than a possibility.

This leaves a set of other ways a buyer-supplier relation may be expressed: Collaborations, joint ventures, and other forms of partnerships do not have an obvious directionality but may still result in dependencies that are relevant for a supply chain map. Furthermore, buyer-supplier relations can be only implied, ambiguous or explicitly stated as uncertain, such as "company A is in talks with company B over the

[8] Flair (`https://github.com/zalandoresearch/flair`; last accessed: 2019-06-11)
[9] Stanford CoreNLP (`https://stanfordnlp.github.io/CoreNLP/index.html`; last accessed: 2019-06-11)

purchase of" or "company A plans to buy from company B". To avoid too many different classes and to ensure that only one class is ever applicable, these cases are grouped into a single third class of buyer-supplier relations (c) that are undirected, or implied or stated as uncertain. This class also aims to ensure that examples of the first two classes are as reliable as possible.

We decided to distinguish a further class of relations (d) where one organisation owns another (fully or partially) or is part of another organisation. Without this class, such relations could be misinterpreted as normal buyer-supplier relations (e.g. "company A buys a stake in company B" or "company A sells its manufacturing business B to company C"). The purpose of this class is less to obtain ownership relations which could be obtained from publicly available reports or databases but to facilitate the annotation decisions and ensure the purity of the other buyer-supplier relationship classes.

Because the named entity recognition was performed automatically as part of the pre-processing of a sentence, errors may occur where a labelled text sequence is not an organisation or was incorrectly segmented. To keep the task complexity manageable and to ensure the identical NER tagging results as a starting point for all annotators, annotators were not asked to rectify incorrect NER tags. Instead, a relation could be classified as 'reject' (e) in that case or other circumstances where an annotator felt incapable of assigning a class.

Finally, the case that none of the above classes are appropriate was captured by a last class (f). This is the most common case for sentences randomly obtained from news articles.

Masking Company names in each sentence were automatically masked so that the classifier did *not* learn relations between *specific* organisational named entities but between any text sequences tagged as organisations. Three types of masks were used: one mask each for the two organisational named entities in question ("__NE_FROM__" and "__NE_TO__"), and one mask ("__NE_OTHER__") for all other organisational named entity mentions not in question but occurring in the sentence. The exact character sequences of these masks are irrelevant; they just need to be uncommon enough to not be confused with any words expected to appear in the input text. As Figure 7 shows, for each possible unordered pair of two organisational named entities, the masking will be different. A sentence with three organisational named entities, for instance, will result in three differently masked versions. For each of these masked versions, the classification algorithm is supposed to consider the relation between the entities masked as "__NE_FROM__" and "__NE_TO__". This way it is ensured that relations between all organisational named entities in a sentence are classified. For a given pair of organisational named entities, it is sufficient to always mask the one mentioned first as "__NE_FROM__" and the one mentioned thereafter as "__NE_TO__". This is because classes are either non-directional or there is a class for each directionality.

Labelling process The labelling was conducted independently by seven annotators who had received the same written instruction as well as an introductory labelling session. To facilitate the labelling, a Web app had been developed to provide an interactive user interface, as shown in Figure 6. For each pair of two organisational named entities that had been automatically detected, the most appropriate relation could be chosen from a drop-down menu.

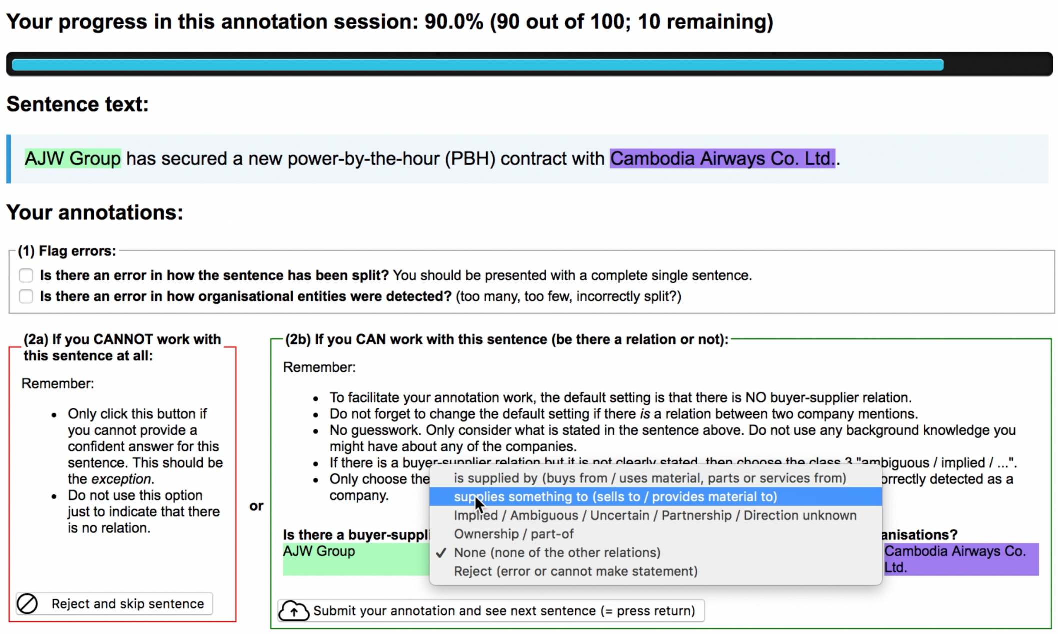

Figure 6. Interface of the annotation app: Each pair of already auto-detected organisational named entities was labelled by human annotators

To measure inter-annotator agreement, a subset of all sentences had to be labelled by all annotators. In addition, within each labelling session of 100 sentences, 5 sentences from the beginning of a session were randomly re-injected towards the end to measure intra-annotator agreement. To obtain the final dataset, a simple majority vote was performed for each organisation-to-organisation relation in each sentence across all annotators. The overall process is illustrated by Figure 7.

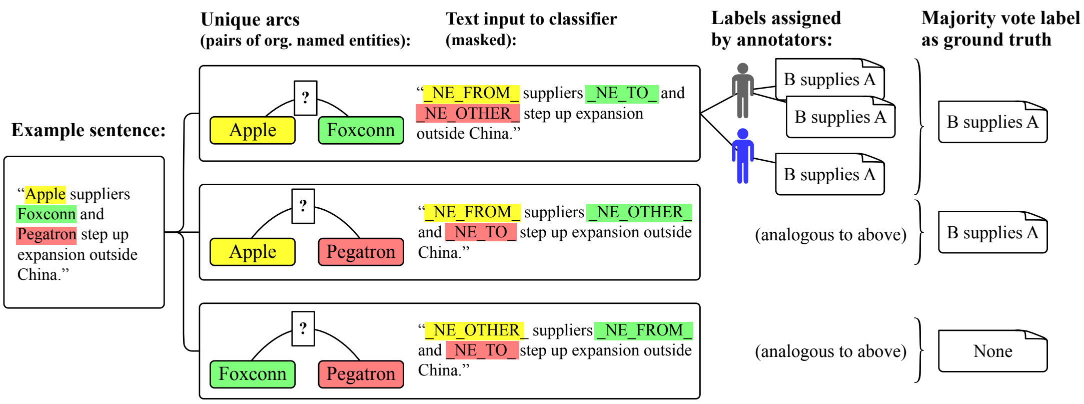

Figure 7. Each sentence may contain multiple pairs of organisational named entities; each pair gets labelled potentially multiple times and potentially by multiple annotators.

The organisational named entities were not masked but revealed to the annotators. This decision was made deliberately to not make the labelling task more difficult by adding a layer of abstraction. To avoid incorrect labels, annotators were specifically instructed not to use the company names as a clue for their labelling decision, to only consider the information provided by the sentence at hand, and to not use any personal background knowledge about the relationship between two organisations.

4.3. *Classification*

Classes For the training and testing of the classifier, the classes 'no buyer-supplier relation' and 'rejected by annotator' were merged as any application based on the classifier would likely treat the 'reject' class the same as a non-existing relation, especially in case of incorrectly identified organisations. This results in *5 classes* that need to be distinguished by the classifier.

Used algorithms In the domain of Machine Learning and NLP, new network architectures are published frequently and the state-of-the-art is a fast-moving target. As a representative of the current state-of-the-art, a BiLSTM deep neural network was chosen. To add further points of comparison, we also chose a multi-layer perceptron (MLP) as well as a linear Support Vector Machine (SVM) classifier (Boser, Guyon, and Vapnik 1992).

Baseline performance To establish a baseline performance, we use two dummy classifiers: a random one and a stratified one. As the name suggests, the random dummy classifier votes fully randomly, resulting in a uniform distribution of assigned labels. The stratified baseline classifier[10] votes randomly but respects the training set's class distribution. That is if the class "None" represents 70% of all assigned class labels, then the baseline classifier would vote "None" randomly but 70% of the time. The stratified classifier is expected to outperform the fully random classifier.

Details on the neural network classifiers (MLP and BiLSTM) The *feature set* is identical for both neural network classifiers and is based on the word embeddings obtained from the GloVe dataset. The dataset, originally consisting of 840B tokens and 300-dimensional vectors trained on Common Crawl, was filtered by those tokens actually present in the training data. Each mask was assigned a separate embedding vector, e.g. the mask "__NE_FROM__" was assigned the 300-dimensional vector $\begin{pmatrix}1 & \dots & 1 & 1 & 1 & 0\end{pmatrix}$. The other masks were assigned the vectors $\begin{pmatrix}1 & \dots & 1 & 1 & 0 & 1\end{pmatrix}$ and $\begin{pmatrix}1 & \dots & 1 & 0 & 1 & 1\end{pmatrix}$, respectively.

The BiLSTM and the MLP architecture were designed to expect 380 features as input. This means that, for each classification task, a text sequence of up to 380 "words" can be fed into the network. Each "word" is represented by its corresponding 300-dimensional embedding. The BiLSTM has an embedding layer and considers 16 features in the hidden state of the LSTM layer. It also uses a dropout with a probability of 0.5. The last layer is a dense layer with 5 output units, one for each class. The LSTM model used the standard hyperbolic tangent function ("tanh") activation function. The MLP architecture is identical in terms of input and output. An embedding layer is followed by a single hidden layer of size 128. This then connects to the output layer of size 5. In initial tests of increasing the network depth did not lead to noticeable performance improvements. ReLU was used as an activation function for the MLP. In the case of both neural network classifiers, a Softmax layer is used to

[10]Stratified dummy classifier provided by the scitkit-learn library (`https://scikit-learn.org/stable/modules/generated/sklearn.dummy.DummyClassifier.html`; last accessed: 2019-06-13)

normalise the outputs so that they can be interpreted as probabilities. As common for single-label multiclass classification problems, categorical cross-entropy was used as a loss function for the neural networks. "Adam" (Kingma and Ba 2014) was used as optimisation algorithm. Each network is trained and tested multiple times and the results are averaged over all runs.

Details on the linear SVM classifier Generally, an SVM is a discriminative classifier that estimates a separating hyperplane in a high-dimensional feature space given labelled training data. The algorithm outputs an optimal hyperplane which can be used to categorise new examples. We use a grid search to tune the hyperparameter of the SVM classifier that is commonly referred to as C. Simply speaking, SVMs aim to fit a hyperplane to separate data points such that (1) the largest minimum margin between different classes is achieved and (2) as many instances as possible are correctly separated. As it is not always possible to optimise both, the C parameter determines the importance of (1).

For the SVM model, a different data representation had to be chosen. We use a simple bag-of-words approach (Joachims 1998), where the order of words is disregarded, and a one-hot-vector is used to represent a sentence. A one-hot-vector is a vector with a the length equal to the vocabulary size of the training dataset (in our case 10,803 tokens), a value of '1' is assigned to to the index of the vector if a word appears in the given sentence, '0' otherwise. The SVM does not consider word order nor does it consider positional information about the organisational named entities. Similar to the other algorithms, the SVM is *not* provided with the company names. The SVM classifier is trained on this representation using a *one-vs-rest* setup.

5. Results and discussion

In this section, we present and discuss the results obtained by the approach we proposed in Section 4.

5.1. *Corpus creation*

Characteristics of the text corpus A single sentence may contain more than two mentions of organisational named entities, and thus multiple potential relations. Each unique set of two mentions of organisational named entities in a sentence required a label that describes the "arc" between them. For this paper, we used a dataset of 3,887 annotated unique sentences resulting in 8,231 labelled unique arcs. Roughly half of the sentences come from the randomly sampled pool of sentences and another half from the pool of sentences that has been manually collected. Each unique arc can be labelled redundantly by multiple annotators (inter-annotator agreement) and even by the same annotator (intra-annotator agreement). Thus, the number of assigned class labels (14,632) is higher than the number of unique arcs. The contribution of labels across annotators varied and is shown in Figure 8.

As expected, the resulting dataset is imbalanced: Nearly 70% of assigned labels were "none" (~60%) or "reject" (~10%). The label distribution after majority vote are shown in Figure 9. Because of the imbalance, F_1 score (as opposed to accuracy) is considered the metric to optimise for.

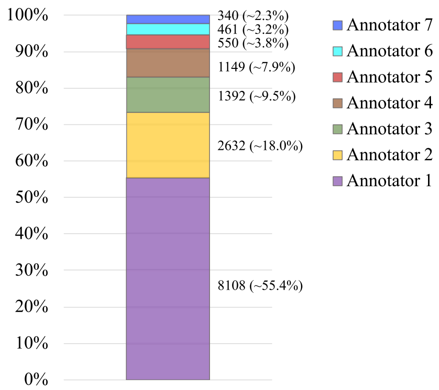

Figure 8. Label distribution by annotator before majority vote (N=14,632 assigned labels)

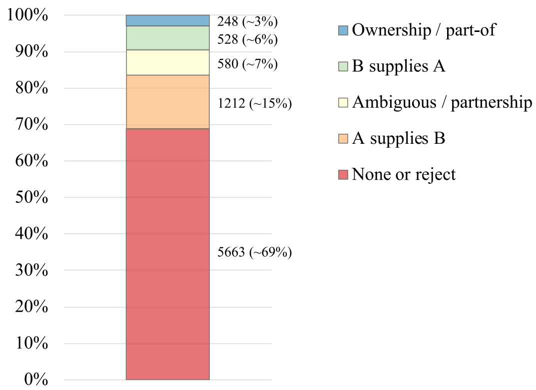

Figure 9. Label distribution for 8,231 arcs (after majority vote)

Achieved inter- and intra-annotator agreement Cohen's κ statistic (Cohen 1960) was chosen to measure annotator agreement. To adapt the metric from a pairwise comparison to more than two annotators, the arithmetic average of Cohen's κ across all pairs of annotators can be computed. The achieved average *inter*-annotator agreement is $\kappa = 0.90$. The average *intra*-annotator agreement is $\kappa = 0.86$. Values of both metrics suggest annotations of good quality.

5.2. *Classification*

The achieved classification results are shown in Table 1. Given the class imbalances, we report the micro-averaged metrics instead of macro-averaged ones. A macro-averaged metric would initially be computed independently for each class and then averaged over all classes. This would treat all classes equally despite their different sizes. Furthermore, in multi-class single-label scenarios, the micro-averaged recall equals the micro-averaged precision, and hence the F_1 score. Oversampling the minority classes was conducted but did not visibly improve classification performance. The dataset was partitioned into training set (70%), validation set (10%), and test set (20%). The neural networks were implemented in Python 3.6 using PyTorch and trained on a single Linux desktop machine using an NVidia GeForce GTX 1080 Ti GPU. Using above system and dataset, a single training and testing run (e.g. BiLSTM trained over 22 epochs and using a batch size of 32) could be completed in approximately one minute. As is common practice, the loss, and the F_1 score on the validation data were observed while increasing the number of epochs to avoid over- or underfitting. The model with the best score was automatically saved to avoid under- or overfitting with respect to the number of epochs. The training was conducted up to 100 epochs, and in an initial trial up to 1000 epochs. With a batch size of 32, the best BiLSTM model in our tests was obtained between epoch 17 and 37.

The overall results are provided by Table 1. As expected, the stratified dummy classifier outperforms the random one and achieves a micro-averaged F_1 score of 0.38. The actual classifiers perform well-above this baseline.

Table 1. Classification results

Method	Configuration	F_1 score (micro-averaged)
Random dummy classifier	Fully random dummy baseline classifier (uniform assignment of class labels)	0.20
Stratified dummy classifier	Stratified dummy baseline classifier (random voting respecting the training set's class distribution)	0.38
SVM	Bag-of-words converted into one-hot-vector (word order and position of organisational named entities are not considered)	0.68
MLP	GloVe embeddings; input sequence length of 380; batch size of 32	0.71
BiLSTM	GloVe embeddings; input sequence length of 380; batch size of 32	0.72

The class-wise classification performance for the SVM is shown in Table 2.

Table 2. Classification results per class – SVM

	Accuracy	Precision	Recall	F_1 score
Class 0: None or reject	0.73	0.80	0.82	0.81
Class 1: B supplies A	0.92	0.39	0.26	0.31
Class 2: A supplies B	0.82	0.38	0.42	0.40
Class 3: ambiguous/undirected	0.92	0.35	0.35	0.35
Class 4: ownership/part-of	0.97	0.47	0.35	0.40
Micro-averaged				**0.68**

The class-wise classification performance for the MLP is shown in Table 3.

Table 3. Classification results per class (averaged over 10 runs) – MLP

	Accuracy	Precision	Recall	F_1 score
Class 0: None or reject	0.78	0.82	0.87	0.85
Class 1: B supplies A	0.91	0.30	0.22	0.26
Class 2: A supplies B	0.84	0.44	0.44	0.44
Class 3: ambiguous/undirected	0.92	0.34	0.36	0.35
Class 4: ownership/part-of	0.97	0.69	0.18	0.28
Micro-averaged				**0.71**

The class-wise classification performance for the BiLSTM is shown in Table 4.

Table 4. Classification results per class (averaged over 10 runs) – BiLSTM

	Accuracy	Precision	Recall	F_1 score
Class 0: None or reject	0.78	0.85	0.85	0.85
Class 1: B supplies A	0.92	0.33	0.21	0.25
Class 2: A supplies B	0.83	0.42	0.63	0.51
Class 3: ambiguous/undirected	0.93	0.42	0.24	0.31
Class 4: ownership/part-of	0.97	0.58	0.22	0.31
Micro-averaged				**0.72**

Even though, the BiLSTM achieved a micro-averaged F_1 score of approximately 0.72 compared to 0.71 achieved by the MLP, this shall not suggest that the BiLSTM is generally the best algorithm for the problem at hand. Despite multiple runs, differences may still be due to chance and not all algorithms, configurations and data representations could be tested. The trained classifiers appear to be able to distinguish well between Class 0 and all others, as demonstrated by the F_1 score of 0.85 for Class 0 achieved by both neural network architectures. Especially, recall of the Classes 1 to 4 still remains relatively low which is likely due to the small size of the obtained dataset. More concretely, the classifier may encounter completely new linguistic expressions in the test phase that it did not encounter during the training phase. This is true in particular for the classes with small sample size, such as "ownership / part-of" and "B supplies A". Regarding the interpretation of the achieved results, the following limitations have to be considered:

The obtained dataset focussed on two manufacturing industries, automotive and aerospace, which may limit its usefulness for other industries with different supply network structures. While some generic expressions, such as "supplies with", are used across industries and the classifier should perform well for these, other expressions may be more industry-specific and including examples of these in the dataset could lead to the discovery of further relations.

Because the human annotation was collected for already NER-tagged sentences, the error introduced by the NER itself is not considered in the stated classification performance.

With regards to defining relationship classes, there appears to be a trade-off between the number of relationship classes and the simplicity of the classification task. More classes may lead to a longer annotation time or lower labelling quality. On the other hand, the defined classes are still limited in their ability to distinguish more subtle semantic differences in relationships. For example, it may be useful to distinguish buyer-supplier relations that have just ended from those who explicitly never existed, and a relationship that is explicitly said to have never existed may have to be distinguished from one where information is just lacking.

It seems as if much of the information is encapsulated in the words themselves rather than the word order. However, it should be immediately clear that a model that does not consider word sequences cannot possibly always correctly distinguish the class "A supplies B" from the class "B supplies A". For some of those sentences, the set of words is identical and only the order of the entity mentions in the sentence is different. Thus, it is expected that models able to consider sequences (sequence models) will outperform those models that are not able to do so.

6. Visualising relations in a basic supply chain map

To obtain a basic supply chain map from the set of relation triples, two simple aggregation steps can be performed to achieve a minimal level of aggregation by deduplicating entity mentions and relation occurrences.

Collapsing identical entity mentions: By feeding in the set of triples into visualisation tools, such as D3.js[11] or Cytoscape[12], entity mentions with identical names will be collapsed into one, as shown in Figure 10. This step is an implicit, naïve form of entity disambiguation where entity mentions with identical names are assumed to refer to the identical entity, and entity mentions with different names are assumed to refer to

[11] d3.js (`https://d3js.org/`); last accessed: 2019-06-11
[12] Cytoscape (`https://cytoscape.org/`); last accessed: 2019-06-11

different entities. For instance, "Toyota Motor Corporation" and "Toyota" would be considered two separate entities.

Collapsing repeated relation occurrences: A further aggregation step is to treat multiple occurrences of the same relation in different sentences or documents as an attribute of the relation. This attribute is visualised not as separate links but, for instance, as the width or colour of a link. This is illustrated in Figure 11 where the number of relation occurrences is also indicated by the link width and a number on the link.

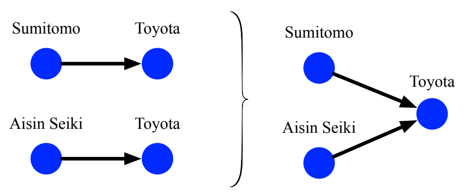

Figure 10. Collapsing nodes

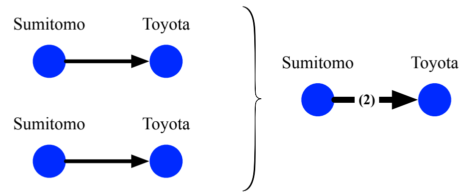

Figure 11. Collapsing repeated relation occurrences

Figure 12 shows a simple, automated visualisation of automatically extracted buyer-supplier relations in form of a basic supply chain map. This particular visualisation was implemented in d3.js. Different relation classes are represented by different line types, e.g. the ownership relation is represented by a dashed line. Directional relations are visualised using arrows pointing in the direction of the material flow.

To obtain the map, the following authentic sentences were processed: "ASCO manufactures and supplies Toyota with these water pumps. ASCO, manufacturer of high lift device mechanisms, complex mechanical assemblies and major functional components, signed a long term contract with Airbus for the production of hybrid complex frames. Denso supplies Toyota with approximately half of its components. GKN Aerospace has been awarded a contract by Airbus. Velocity Composites has signed a new contract that will see it supply aerospace manufacturer GKN Aerospace with structural plies for the next five years. Japanese car brands Toyota and Suzuki have announced wide-ranging global collaboration plans. Toyota owns close to 25% of subsidiary Denso." Relations that were identified as directed ones are indicated as such in the map. The arrow head indicates the detected material flow. The classifier interpreted the relation between Toyota and Suzuki as non-directional / partnership, as indicated by the lacking arrow head for this link. The input text described two relation types between Toyota and Denso: a directed buyer-supplier relation and an ownership one. The size of this knowledge graph can become arbitrarily large by processing additional news data.

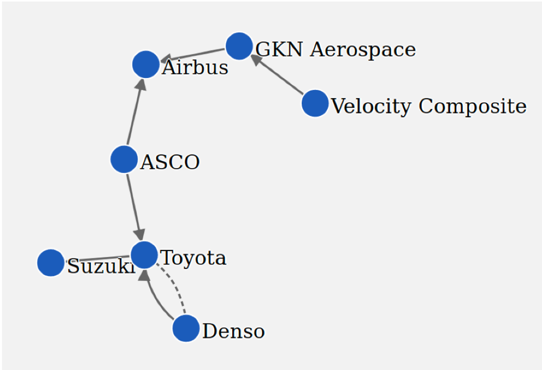

Figure 12. Basic supply chain map based on extracted relations

It is obvious that supply chain maps that can be generated solely based on simple company-to-company relations are limited. For instance, the visualisation appears to suggest that Velocity Composites is a sub-tier supplier of Airbus. However, the provided text example alone does not provide sufficient evidence for this inference. One way to address this "transitivity problem" is to also extract the end-product for which a part is intended if this fact is mentioned in the context. The visualisation can also be further enriched. E.g. the confidence in each relation classification could be indicated.

7. Conclusion & future research

To address the problem of limited visibility of extended supply chain structures, we proposed to automate the extraction of supply chain maps from news articles using Natural Language Processing and Machine Learning technology. A fundamental building block for this approach is the extraction of individual buyer-supplier relations between two organisations from natural language text. The contributions of this paper are thus the following: We first proposed a methodology for obtaining a text corpus to evaluate the performance of classifiers that are designed to extract buyer-supplier relationships. Following the methodology, we were able to show that an inter-annotator agreement of 0.90 is possible by obtaining a corpus that can be used to train and test classifiers, such as neural networks. Furthermore, we proposed an approach to convert example sentences into feature vectors by masking the names of organisational entities. Lastly, we were able to obtain a first baseline classification performance for buyer-supplier relations: A micro-averaged F_1 score of >0.7 suggests that the automated extraction is indeed a viable path forwards. The generated triples can be visualised in a basic supply chain map.

The classifier can be further improved by adding more training examples following the same procedure described in this paper. Furthermore, models based on even more recent NLP developments, such as so-called transformers using the attention mechanism, could be alternative options. Having a trained model allows us to fully automatically extract buyer-supplier relations from large unlabelled text corpora and to visualise the extracted buyer-supplier relation in a network. Thus, in future work, we would like to apply the trained classifier to a large unlabelled dataset to be able to answer questions regarding the availability and density of information, especially in varying industrial contexts. In a first test, the trained classifier applied to the Reuters TRC2 dataset returned about 37,000 instances of company-to-company relations that were predicted to be buyer-supplier ones (Classes 1 to 3). To improve precision, these examples – as opposed to a sparse dataset of random samples – can be manually labelled and then added to the training dataset.

The value of the generated supply chain maps could further be increased by incorporating additional tasks. Entity disambiguation and entity linking would allow to provide additional information, such as the geolocation, industry or size of a company. The information extraction can also be extended to cover provided goods and services as well as the intended end-product of those goods and services. Recently, massive pre-trained language models, such as GPT-2, BERT and others, have been made available and offer a further exciting future research perspective. The knowledge captured by being trained on extremely large corpora could potentially also be leveraged for the extraction of buyer-supplier relations or even the prediction of potential suppliers. Generally, the approach proposed in this paper could help reduce risks associated with limited visibility of multi-tier supply networks by complementing existing supply chain mapping efforts.

References

Achilles Group. 2013. "Supply Chain Mapping." Accessed 2016-08-30. http://www.achilles.co.uk/images/pdf/communitypdfs/Automotive/Achilles-Supply-Chain-Mapping.pdf.

Bach, Nguyen, and Sameer Badaskar. 2007. "A review of relation extraction." *Literature review for Language and Statistics II* 2.

Barratt, Mark, and Adegoke Oke. 2007. "Antecedents of supply chain visibility in retail supply chains: A resource-based theory perspective." *Journal of Operations Management* 25 (6): 1217–1233.

Basole, Rahul C., and Marcus A. Bellamy. 2014. "Supply Network Structure, Visibility, and Risk Diffusion: A Computational Approach." *Decision Sciences* 45 (4): 753–789.

Boser, Bernhard E, Isabelle M Guyon, and Vladimir N Vapnik. 1992. "A training algorithm for optimal margin classifiers." In *Proceedings of the fifth annual workshop on Computational learning theory*, 144–152. ACM.

Business Continuity Institute. 2014. "Supply Chain Resilience 2014 Annual Survey." http://www.bcifiles.com/supply-chain.pdf.

Choi, Thomas Y., and Yunsook Hong. 2002. "Unveiling the structure of supply networks: Case studies in Honda, Acura, and DaimlerChrysler." *Journal of Operations Management* 20 (5): 469–493.

Chollet, Francois. 2018. *Deep Learning with Python*. Manning Publications Co.

Chowdhury, Gobinda G. 2003. "Natural language processing." *Annual review of information science and technology* 37 (1): 51–89.

Christopher, Martin, and Hau Lee. 2004. "Mitigating Supply Chain Risk Through Improved Confidence." *International Journal of Physical Distribution & Logistics Management* 34 (5): 388–396. http://dx.doi.org/10.1108/09600030410545436.

Christopher, Martin, and Helen Peck. 2004. "Building the resilient supply chain." *International Journal of Logistics Management* 15 (2): 1–13. http://dx.doi.org/10.1108/09574090410700275.

Cohen, Jacob. 1960. "A coefficient of agreement for nominal scales." *Educational and psychological measurement* 20 (1): 37–46.

Cowie, Jim, and Wendy Lehnert. 1996. "Information extraction." *Communications of the ACM* 39 (1): 80–91.

Farris, M. Theodore. 2010. "Solutions to strategic supply chain mapping issues." *International Journal of Physical Distribution & Logistics Management* 40 (3): 164–180.

Gardner, John T, and Martha C Cooper. 2003. "Strategic Supply Chain Mapping Approaches." *Journal of Business Logistics* 24 (2): 37–64.

Goh, Mark, Robert De Souza, Allan N. Zhang, Wei He, and P. S. Tan. 2009. "Supply chain visibility: A decision making perspective." *2009 4th IEEE Conference on Industrial Electronics and Applications, ICIEA 2009* 2546–2551.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. 2016. *Deep learning*. 1st ed. Cambridge, MA, USA: The MIT Press.

Graves, Alex, and Jürgen Schmidhuber. 2005. "Framewise phoneme classification with bidirectional LSTM and other neural network architectures." *Neural Networks* 18 (5-6): 602–610.

Hearst, Marti A. 1992. "Automatic Acquisition of Hyponyms from Large Text Corpora." *Proceedings of the 14th conference on Computational Linguistics* 2: 23–28.

Hochreiter, Sepp, and Jürgen Schmidhuber. 1997. "Long short-term memory." *Neural computation* 9 (8): 1735–1780.

Ivanov, Dmitry, Boris Sokolov, and Alexandre Dolgui. 2014. "The Ripple effect in supply chains: trade-off efficiency-flexibility-resilience in disruption management." *International Journal of Production Research* 52 (7): 2154–2172. `https://doi.org/10.1080/00207543.2013.858836`.

Joachims, Thorsten. 1998. "Text categorization with support vector machines: Learning with many relevant features." In *European conference on machine learning*, 137–142. Springer.

Jüttner, Uta, Helen Peck, and Martin Christopher. 2003. "Supply chain risk management: outlining an agenda for future research." *IJ of Logistics: Research and Applications* 6 (4): 197–210.

Kingma, Diederik P, and Jimmy Ba. 2014. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980* .

KPMG International & The Economist Intelligence Unit. 2013. "The Global Manufacturing Outlook 2013 - Competitive advantage: enhancing supply chain networks for efficiency and innovation." 27. `http://www.kpmg.com/DE/de/Documents/global-manufacturing-outlook-2013-kpmg.pdf`.

Lambert, Douglas M, and Martha C Cooper. 2000. "Issues in Supply Chain Management." *Industrial Marketing Management* 29 (1): 65–83.

Maedche, Alexander, and Steffen Staab. 2001. "Ontology Learning for the Semantic Web." *IEEE Intelligent Systems* 72–79.

Mentzer, JT John T., William Dewitt, JS James S Keebler, Soonhong Min, Nancy W Nix, Carlo D Smith, and Zach G Zacharia. 2001. "Defining supply chain management." *Journal of Business Logistics* 22 (2): 1–25. `http://onlinelibrary.wiley.com/doi/10.1002/j.2158-1592.2001.tb00001.x/abstract`.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. "Efficient estimation of word representations in vector space." *arXiv preprint arXiv:1301.3781* .

Mintz, Mike, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. "Distant supervision for relation extraction without labeled data." *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - ACL-IJCNLP '09* 2 (2005): 1003. `http://portal.acm.org/citation.cfm?doid=1690219.1690287`.

Murphy, Kevin P. 2012. *Machine Learning: A Probabilistic Perspective*. The MIT Press.

Nadeau, David, and Satoshi Sekine. 2007. "A survey of named entity recognition and classification." *Lingvisticae Investigationes* 30 (1): 3–26.

Pyysalo, Sampo, Simon Baker, Imran Ali, Stefan Haselwimmer, Tejas Shah, Andrew Young, Yufan Guo, et al. 2018. "LION LBD: a Literature-Based Discovery System for Cancer Biology." *Bioinformatics* .

Sheffi, Yossi. 2005. *The Resilient Enterprise: Overcoming Vulnerability for Competitive Advantage*. Vol. 1. The MIT Press.

Soto-Viruet, Yadira, W. David Menzie, John F. Papp, and Thomas R. Yager. 2013. *USGS Open-File Report 20131239 - An Exploration in Mineral Supply Chain Mapping Using Tantalum as an Example*. Technical Report. `http://pubs.usgs.gov/of/2013/1239/`.

Suarez-Barraza, Manuel F., José Miguel-Davila, and C. Fabiola Vasquez-García. 2016. "Supply chain value stream mapping: a new tool of operation management." *International Journal of Quality and Reliability Management* 33 (4): 518–534.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. "Attention is all you need." In *Advances in neural information processing systems*, 5998–6008.

Wichmann, Pascal, Alexandra Brintrup, Simon Baker, Philip Woodall, and Duncan McFarlane. 2018. "Towards automatically generating supply chain maps from natural language text." *IFAC-PapersOnLine* 51 (11): 1726 – 1731. 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018, `http://www.sciencedirect.com/science/article/pii/S2405896318313284`.

Yamamoto, Ayana, Yuichi Miyamura, Kouta Nakata, and Masayuki Okamoto. 2017. "Company Relation Extraction from Web News Articles for Analyzing Industry Structure." In *Semantic Computing (ICSC), 2017 IEEE 11th International Conference on*, 89–92. IEEE.

Yan, Tingting, Thomas Y Choi, Yusoon Kim, and Yang Yang. 2015. "A theory of the nexus supplier: a critical supplier from a network perspective." *J of Supply Chain Management* 51 (1): 52–66.