The influence of knowledge in the design of a recommender system to facilitate industrial symbiosis markets

Guido van Capelleveen[a], Chintan Amrit[a], Devrim Murat Yazan[a], Henk Zijm[a]

[a] *Department of Industrial Engineering and Business Information Systems, University of Twente, The Netherlands*

Abstract

Industrial symbiosis (IS) is a business tool engaging cooperation among industrial firms to utilize industrial waste streams from other industries and to share IS knowledge, in order to achieve sustainable production. IS recommenders can support industries through the identification of item opportunities in waste marketplaces, enhancing activity that may lead to the development of an active waste exchange network. To build effective recommendation, we study the role of knowledge in the design of a recommender that suggests waste materials to processing industries. This paper compares the performance of a knowledge-based input-output recommender with a recommender based on association rules. The two recommenders are evaluated with real-world data collected through deploying surveys in a workshop setting. Our research shows that many data challenges arise when creating recommendations from explicit knowledge and suggests that techniques based on the concept of implicit knowledge are preferred in the design of an IS recommender.

Keywords: Industrial Symbiosis, Recommender systems, Decision Support Systems, Input-output matching, Association-rule mining

[*] Corresponding author at: University of Twente, PO box 217, 7500 AE Enschede, The Netherlands, E-mail address: g.c.vancapelleveen@utwente.nl (Guido van Capelleveen)

Preprint submitted to Environmental Modelling and Software journal *March 9, 2017*

1. Introduction

Reduction of waste emissions and primary resource use in resource-intensive industries are suggested as one of the ways to contribute towards more sustainable development (European Environmental Agency, 2016). Industrial symbiosis (IS) can contribute to that through the identification and utilization of traditional secondary production process output in an organization, considered as waste. The waste can be exchanged to substitute (part of) a primary resource, possibly after some preprocessing, of a production process of another organization, usually located in a different industrial sector (Chertow, 2000). IS has developed as a useful business opportunity and tool for eco-innovation and has led to a proposal for practitioners that emphasizes the practical context. It involves ”engaging diverse organizations in a network to foster eco-innovation and long-term culture change, creating and sharing knowledge through the network yielding mutually profitable transactions for novel sourcing of required inputs, value-added destinations for non-product outputs, and improved business and technical processes” (Lombardi and Laybourn, 2012). The implementation of new exchanges is mostly facilitated by industry workshops (Paquin and Howard-Grenville, 2012; Van Beers et al., 2007; Mirata, 2004), industrial symbiosis identification systems (Grant et al., 2010), and waste exchange marketplaces (Dhanorkar et al., 2015) through which information of the waste and resource interests are exchanged.

The paths leading to the emergence of IS are researched by various scholars pointing out different stylized models of IS (Chertow, 2007; Paquin and Howard-Grenville, 2012). Eco-industrial parks generally involve a continuous effort among coordinating bodies, e.g. municipalities or regional governments, to locate potentially interesting industries closer together in park regions in order to share wastes and by-products (Gibbs and Deutz, 2007). Other industrial ecosystems are self-organizing, resulting from collaborations without top-down planning and mainly driven by economical or strategic business motivations that lead to increasing resource and waste transactions over time (Chertow and Ehrenfeld, 2012). In between, a facilitated approach is considered as a third type of IS emergence that utilizes intermediaries that provide a role of strengthening trust between firms using expertise and the ability to connect industries (Paquin and Howard-Grenville, 2012). These pathways characterize the different types of emergence and explain how the process of IS unfolds, from which one can deduce the critical catalyzers to initiate new symbiotic actions (Boons et al., 2016).

In relation to the facilitated approach, various scholars have studied waste-exchange systems as a tool to enhance IS identification (Clayton et al., 2002; Sterr and Ott, 2004; Mirata, 2004; Chen et al., 2006; Van Beers et al., 2007; Veiga and Magrini, 2009; Grant et al., 2010; Dietrich et al., 2014; Dhanorkar et al., 2015; Cecelja et al., 2015; Hein et al., 2015; Cutaia et al., 2015). Such decision support can engage network exchange while substantially reducing the time investment for an investigation of the potential for symbiosis. Decision support tools, and in particular recommender support, can stimulate the identification and assessment of new exchanges. However, building a type of decision support that can recommend IS opportunities remains as a challenge. Often such tools lack the key characteristic of sociability and focus more on determining technical opportunities rather than building human relationships. Furthermore, such tools do not reach the required critical mass of adoption and struggle with the complexity of information exchange required to identify IS (Grant et al., 2010). Process data from manufacturing industries that disclose inputs, outputs and wastes, can be used for the identification of potential synergies and provide input for recommender systems. However, the extend and detailedness by which this data is shared may be hindered by a lack of trust among organizations, because process data might reveal competitive information that organizations want to keep (partly) confidential (Paquin and Howard-Grenville, 2012). Therefore, the provision of detailed process data in a more wider context may increase identification results, but remains a key challenge.

The aim of this paper is to evaluate and understand the influential role of knowledge for the design of an effective waste material recommender in IS marketplaces. We employ an analytical method which empirically evaluates the design of a recommender, based on explicit knowledge, and compare its performance against a recommender based on implicit knowledge, also referred to as tacit knowledge. Here, by explicit knowledge we imply a recommender that uses an external knowledge base, and by implicit knowledge we imply a recommender that learns from data on the behavior of users in a marketplace. Knowing that confidentiality is a key challenge in IS development, we exploit input-output data from external life cycle inventory databases as a knowledge base in the knowledge-based recommender, in order to investigate potential synergies. We utilize association rule mining in order to evaluate the effect of implicit knowledge on the recommendation. This paper presents both the novel design of an Input-Output (IO) algorithm, as well as a comparison of the IO algorithm to an algorithm based on association rule mining. The IS data that is used in the case study contains waste items as well as resource interests, and originates from industrial symbiosis workshops held in two different European countries. Using the results of the experiments, we show that the design of a knowledge-based resource recommender is affected by many data challenges, including linking waste streams to process inputs, structural and semantic representation, attribute availability, code standardization, data reliability and data integrity problems. Furthermore, we observe that implicit knowledge-based algorithms outperform explicit knowledge-based algorithms in identifying IS opportunities. This work contributes by filling a gap in the IS literature, by addressing the difficulties in building practical IS decision support in a facilitated context. Moreover, it assists researchers in the techniques that can improve the quality of IS data, or contribute to IS decision-making.

2. Research design

A review of ICT tools for IS development conducted by Grant et al. (2010) shows that many tools demonstrate technical feasibility but lack sociability and a critical mass of adoption. In addition, the paper emphasizes that many challenges associated with IS identification originate from the fact that the creation of IS depends on tacit knowledge, to a large extent (Grant et al., 2010). Recommenders are able to support users in identifying item opportunities and to pro-actively engage system use, resulting in both increased sales and a more active community (Freyne et al., 2009; Pathak et al., 2010; Gomez-Uribe and Hunt, 2015). To design an effective waste recommender, we investigate the influential role of knowledge in IS markets on recommender design. The empirical method of design evaluation is applied to two recommenders in order to investigate the extent to which explicit and tacit knowledge influences the ability to create recommendations.

The research design consists of four steps:

1. Data is collected from industrial symbiotic workshops. Through data exploration, we review the variety of waste items and then select the item-properties that are valuable in a recommender model.

2. Thereafter, we cluster the ephemeral items in our data set into latent product concepts, based on a technique described in Chen and Canny (2011).
3. Then, a novel design is created for a knowledge-based resource recommender algorithm using explicit knowledge derived from life cycle inventory databases. This method is based on the concept that a waste-to-input match can be predicted using an industrial manufacturing profile. This profile reveals the materials that are used in large quantities for most associated production processes. The design exploits the potential of life-cycle inventory databases providing process data about the inputs, outputs and wastes of an industry in order to construct a resource profile that identifies which type of inputs and outputs are likely to be available. A potential benefit of this knowledge based approach is that it removes the bootstrap problem that most other recommender algorithms in e-marketplaces face (Ekstrand et al., 2011). The design science research methodology (Peffers et al., 2007) guides our work in designing the knowledge-based input-output (IO) algorithm.
4. Finally, the IO based recommender is tested, evaluated and compared with a recommender based on tacit knowledge, using survey data from IS workshops. This alternative recommender incorporates the promising technique of association rule mining often applied in e-marketplaces (Park et al., 2012).

3. Clustering items

One of the prerequisites to many recommender algorithms, in particular to our proposed Input-Output (IO) algorithm and the Association Rule Mining (ARM), is a reduction of the item space. This reduction is required to decrease the computational complexity and increase the item transaction history. A rich item history allows the ARM to deduce more and stronger associations. Moreover, it can improve IO matching on items with small item-descriptions by using a richer product concept description. The key challenge of item space reduction is specifically apparent in e-commerce marketplaces that predominantly consist of ephemeral items composed of users' product descriptions, that are entered by them. Often item descriptions neither correspond to any catalog taxonomy, nor provide detailed product descriptions. Moreover, these marketplaces comprise a volatile product catalog (Wroblewska et al., 2016). Recommender techniques, such as Association

Rule Mining (ARM), attempt to find patterns in transaction data, which in turn requires transactions of identical items. The challenge for this type of recommender is that such data are sparse and the item space needs to be reduced. Hence, grouping similar items helps to increase quality, efficiency and effectiveness of recommender algorithms (Chen and Canny, 2011).

An apparent method to group waste market items is to make use of existing resource or waste taxonomical classifications, that already play an important role in the domain of waste treatment. For example, the European Waste Catalogue (EWC) (European Commission, 2000) or the Central Product Classification (CPC) (United Nations Statistics Division, 2015) are taxonomies used in the legal obligation of waste disposal reporting and in the 'duty of care' documents in waste transfers (Natural Resources Wales et al., 2015) and can define the similarity between items. However, such classifications are often absent in item descriptions. Moreover, these systems classify goods and services in the industry from which they originate, causing an overlap in product concepts within the taxonomy. Thus, such taxonomy fails to relate two similar waste items if the wastes are produced in different industries (The ISDATA project, 2015; Sander et al., 2008). For example, recycled glass can be produced either with uncontaminated glass residues resulting from a glass bottle production facility, or it is possible to utilize glass extracted from construction and demolition waste (International Synergies Ltd., 2016). Thus, designing recommenders for such long-tail marketplaces requires a more determined and stable representation that can be created by mapping items to latent product concepts (Chen and Canny, 2011). By inferring the product concept, we capture the dynamics and diverse item-inventories into a group of items to be considered as similar or identical products to a recommender. In this way, item-clustering reduces complexity as an intermediate step towards enabling recommender algorithms to learn from data in such contexts.

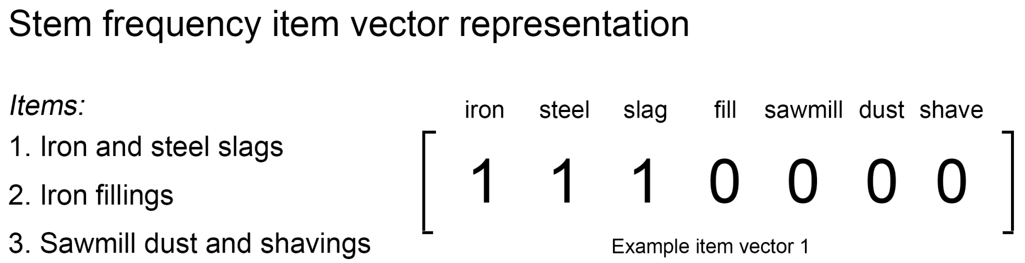
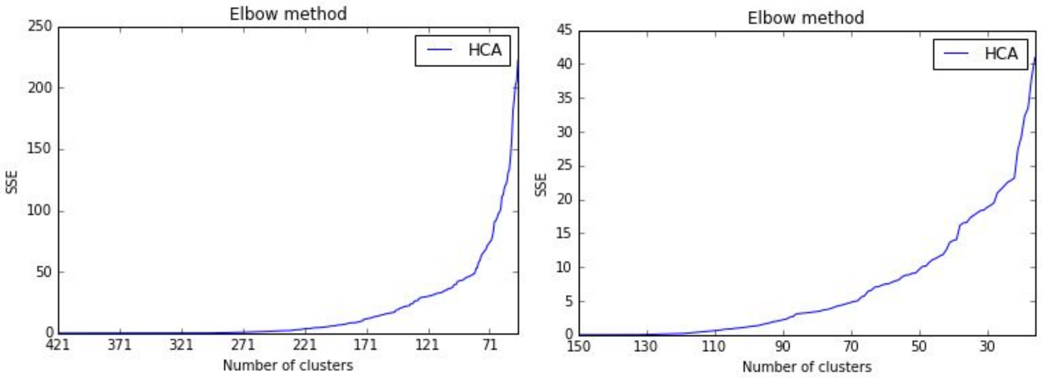


Figure 1: Stem frequency item vector example

Our clustering method that groups similar items builds upon the derivation of a product concepts from item descriptions provided. This leads to an algorithm design that uses stem-frequency vectorization as a means to identify this product concept of an offered waste item (see Algorithm 3 in the Appendix). Waste items addressed the IS workshop survey data often have small waste descriptions, commonly defined with less than 10 words (see Table 1). In some cases additional structured and unstructured item attributes are provided, such as the category, quantity or location of waste. Prior to cluster the items, some data preparation is required, which starts with stemming the item descriptions. Stemming is the process of removing the inflectional forms and sometimes the derivations of the word, by identifying the morphological root of a word (Manning et al., 2008). Creating a frequency vector of an item within a set of item descriptions involves determining the frequency of every unique stem in that item and the unique stems in the set. An example of such item-vector is provided in Figure 1. In order to derive the set of stems from an item description we use the NLTK package, a platform for working with human language data in Python (Natural Language Toolkit, 2017). First, we convert all the characters to lowercase and remove all numbers and special characters. Then, items are tokenized into a bag of words. In these bags, we remove all English stop words from the NTLK corpus and filter some non-significant terminology commonly used in IS, e.g. 'waste', 'material', and 'process'. Finally, we apply the empirically proven Porter algorithm (Porter, 1980) to stem the bag of words. Then, the resulting item vectors are utilized in our proposed multi-dimensional hierarchical agglomerative clustering algorithm to cluster items based on the cosine similarity of item vectors, see Algorithm 4 in the Appendix. In multidimensional clustering, a vector is supplied as a parameter instead of the x,y coordinates in traditional clustering algorithms, which in this case is the stem-frequency vector.

The subsequent step is to define the number of clusters. Often, in hierarchical clustering the elbow method is used to measure the cluster performance by identifying the point of maximum curvature. Our measure of reference to identify statistical variance in order to explain the number of clusters is to calculate the Sum of Squared Errors (SSE) as a function of the decrease in the number of clusters. Then, the 'knee' or 'elbow', the point that shows a marginal drop or gain in variance, defines the number of clusters. Although various statistical methods have been proposed to evaluate all measurements to identify the error in a curve do help (Salvador and Chan, 2004), visual interpretation is preferred over statistical methods when no objective measures have been defined for cluster optimality in the domain of application (Jung et al., 2003). As a result, elbow figures were constructed (see Figure 2), and used to derive the optimal number of clusters for datasets A and B, respectively 84 for region A and 40 for region B.



(a) Dataset A

(b) Dataset B

Figure 2: Sum of Squared Errors (SSE)

4. The input-output algorithm

Knowledge-based recommenders suggest items based on the inferences about the needs and preferences of a user (Burke, 2002). This type of recommender makes use of explicit knowledge (like a knowledge base) to recommend items to users. With this first recommender algorithm we test the applicability of explicit knowledge for predicting waste item preference. The algorithm is based on the assumption that the waste materials offered in IS marketplaces correspond to the specific raw material interest of an organization, and subsequently to the primary raw materials that are used as traditional inputs in their manufacturing processes. As this method uses the identification of potential relations between the primary inputs of production processes and secondary outputs (waste) offered in a marketplace, we refer to this type of raw material recommendation as an input-output (IO) recommendation.

The explicit knowledge of resource-use within the production processes is derived from a Life Cycle Inventory (LCI), such as Ecoinvent (2017), which are commonly used in Life Cycle Assessment (LCA) tools. This type of database provides well documented process data for thousands of production processes, including information on the raw material consumption. In particular, this information about primary inputs is used to identify the most consumed raw materials of an organization and matches those to the potential waste resources available in the marketplace.

Recommendations made by the IO recommender algorithm are created through the use of pre-compiled manufacturing profiles. A profile is constructed through the identification of the type of products an industry produces, typically extracted from the respective company websites. The products are used to find the associated manufacturing processes available from LCI databases. This results in pre-defined profiles created for each type of organization (e.g. a manufacturer of castings). The profiles list, among others, the raw material consumption that is associated with the selected manufacturing processes. To operate the recommender, we create all profiles required to make recommendations to the organizations present in our data set. Once the profiles are created, the algorithm extracts all resources listed from the LCI database that are linked to a production process in the organizational profile. Then, it filters the resources which make up the largest fraction of resource consumption within the production of a product. We select these resources as the candidates for recommendation. The resource candidates are then compared with the marketplace item descriptions to see if they contain similarities. The matching algorithm calculates the cosine similarity between the vectorized stem-frequency of a waste product description (a cluster) and that of a resource description. The stem-frequency vectorization algorithm applied in the clustering of items (explained in Section 3) is identical to the one we use for IO matching. The design of the IO algorithm is illustrated in pseudo-code in Algorithm 1.

Firstly, for each resource connected to the industrial profile of the organization, we iterate over all clusters in the marketplace. Within this iteration, we create a bag of stems for both the selected cluster (our latent product concept) and the selected resource (from our profile). The first bag of stems, created from the cluster, selects the most frequently occurring stems in all items (descriptions) belonging to that cluster. The second bag of stems is created from the resource and uses the name of the material based on the taxonomy used in the LCI. These bags of stems are used to compose the item vectors for both the cluster and the resource. Thereafter, the similarity between the two item vectors is assessed using a cosine similarity function. The stem frequency item vector combinations that exceed the minimum cosine threshold and are not yet a recommendation are added to the set of recommendations.

```
Algorithm 1: Input-output recommender algorithm
Data:
        $N_{resources}$ = Set of associated resources of an organizational
        manufacturing profile derived from LCI
        $W_{items}$ = Set of all waste items in marketplace
        $p_1$ = Minimum cosine similarity
        $B$ = Bag of stems
        $V$ = Stem-frequency item vector
Result:
        R = Set of recommendations
Function Input-Output-recommender($N_{resources}, W_{items}, p_1$)
    foreach $resource \in N_{resources}$ do
        foreach $cluster \in W_{items}$ do
            Create a bag of stems $B_{cluster}$ with the $n$ most frequent stems
              from all items in the $cluster$ where $n$ is the average number
              of stems in each item belonging to that cluster;
            Create a bag of stems $B_{resource}$ from the item $resource$;
            Create a stem-frequency vector $V_{cluster}$ for the $cluster$ with
              $B_{cluster}$ as item;
            Create a stem-frequency vector $V_{resource}$ for the $resource$ with
              $B_{resource}$ as item;
            Calculate the cosines similarity $s$ between cluster vector $V_{cluster}$
              and resource vector $V_{resource}$;
            if $s > p_1$ and $cluster \notin R$ then
                Add a unique recommendation $cluster$ to $R$;
    return $R$;
```

5. The association rule mining algorithm

In this second recommender we attempt to make use of implicit knowledge to predict the preference of items to users. Association Rule Mining (ARM) is one of such techniques that found successful application in recommender systems (Park et al., 2012). To create a recommendation using ARM, we conduct an analysis that attempts to discover regularities in transaction data based on the concept of strong rules (Agrawal et al., 1993). In general, association rules perform at best in large scale data with a broad history of transactions. Often, the popular A priori algorithm is used to generate candidates for identifying these rules, as it is simple and exact (see Agrawal and Srikant (1994)). A drawback of this breadth-first search in the A priori algorithm, is that the iterative calculation of k+1 candidates from the generated and tested candidates can be computational expensive for increasing large candidate sets (Han et al., 2000). This has led to other more efficient algorithms designed for real-life large scale recommenders, such as TP-growth (Zheng et al., 2001). However, for our calculation we select apriori as here we value the completeness of frequent items set over the performance and memory consumption of pair creation. With growing size of transactions it would be best to consider TP-growth because of its scalability (Alfaro and Solano, 2015).

Furthermore, a time frame must be determined considering the transactions from which association rules are learned. The session we select determines if we include only items purchased together or whether multiple transactions within a given period can be considered as one transaction from which associations are mined. In some application contexts, if the decay of rules may be considered less severe, widening the time window can have a positive effect on the number of rules that can either be extracted from the transactions, and on the effectiveness of the predictors. The nature of IS is characterized by a low frequency of transactions, but transactions keep their merit over months enabling one to deduce user behavior valuable to predict new items. Therefore, in the context of IS, it is reasonable to select a wider time window to retrieve association rules, rather than a single transaction. Such a time window may range from several months up to years.

A simple way to define a recommendation, as used in this research, is to set a threshold for the minimal support and confidence of association rules that are used to predict items. In order to be able to test the ARM algorithm on the relatively small data set that was collected, we apply k-fold cross validation during the evaluation of association results, so that all items are considered as a candidate for recommendation to an organization. In this way, the results from both the IO algorithm and the ARM algorithm can be compared with a higher level of validity. This results in the application of the k-fold cross validation and ARM technique presented in the pseudo-code Algorithm 2.

Algorithm 2: (k-fold) association rule mining recommender

```
Data:
        $W_{items}$ = Set of all waste items in marketplace
        $p_1$ = Minimum support
        $p_2$ = Minimum confidence
        $p_3$ = k-value of k-fold validation
        $C$ = Candidate rules
        $I_{training}$ = List of training samples
Result:
        $R$ = Recommendations for each train/test sample
        $I_{test}$ = List of test samples
Function $k$-$fold$-$ARM(W_{items}, p_1, p_2, p_3)$
    for $i \leftarrow 1$ to $p_3$ do
        Divide $W_{items}$ into $p_3$ equally sized samples $s$;
        Assign all samples $s$, except where $s = i$, to training set $I_{training}$;
        Assign sample $s$ where $s = i$ to test set $I_{test}$;
    foreach sample $s \in I_{traning}$ do
        Calculate the candidates $C_{candidates}$ and the support $C_{support}$ from the sample $s$ with a minimum support of $p_1$;
        Generate the association rules $r$ from the candidates $C_{candidates}$ and support data $C_{support}$ with a minimum confidence $p_2$;
        Add the rules $r$ to the rules;
    return $R, I_{test}$;
```

6. Results

6.1. Data exploration and preparation

The data set used to test our algorithms was collected through Industrial Symbiosis (IS) workshops organized to facilitate the creation of new IS initiatives among industries. The workshops were held in two different European regions and record waste items of participating organizations and their expressed interest. The workshop data, although captured electronically, are treated as marketplace transactions. We consider the survey data as representative to design a recommender algorithm, as the data characteristic that users compose their own item descriptions is shared, which is a typical key characteristic in many other electronic marketplaces providing IS sales services.

This workshop data consists of two data sets with a sample size of 421 for industry region A, and 150 for region B. This sample size is relatively low compared to the evaluation of recommenders in many other types of e-marketplace studies, as IS waste markets do not have such large participation. Thus, careful interpretation of the statistical validity of the recommender performance is needed. In particular, one needs to keep in mind that there is a distinct possibility of overfitting the model of the ARM algorithm. However, we derive some qualitative insights into the applicability and challenges of recommender design in IS marketplaces.

A wide variety of waste items are represented in the IS workshop data. We recognize five different classifications of items, namely; (1) Materials, (2) Tools, (3) Services, (4) Energy, and (5) Others (see Figure 3). The IO recommender is designed to predict the material preference of an organization, which can be associated to items in the marketplace. Therefore, the data practicable to evaluate the IO recommender is a selection of 'have' items categorized as material. Therefore, this group is selected to test both the IO-recommender and the ARM recommender. A few of these typical workshop items from our data set are listed in Table 1. The IO recommender algorithm is perceived useful for organizations in the process industry that manufacture goods. Therefore, in recommending items, organizations that do not fit the profile of a manufacturer were excluded from the recommendation.

Table 1: Typical workshop items

Example	Description
a	"Iron and steel slag: Concrete tiles can be taken as one of the main components"
b	"Natural mineral waste"
c	"Chip (Aluminum, Iron, Bronze): Wastes from the manufacturing process"
d	"Waste oil: Oils and household waste oils in region"
e	"Sawmill dust and shavings"

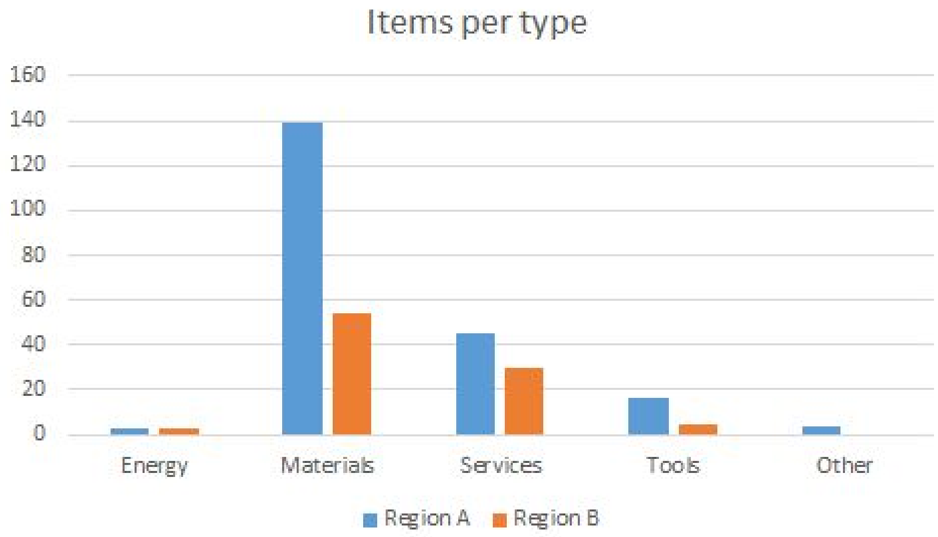


Figure 3: Items per type

6.2. Algorithm settings

The experimental setup of this research is determined by a couple of parametric settings in the algorithms used. First, in order to determine the number of clusters for each data set we use the elbow figure to derive a small k value with a low SSE. Following the optimal k (see Section 3), we assign a 'cluster' label to each of the workshop items. The remaining parameters are consistent over both data sets. In the input-output algorithm we use a minimal cosine similarity of 0.1, reflecting that one of the stems derived from the item description matches the resource taxonomy of the Life Cycle Inventory (LCI). This value needs to be kept low, as nearly all item descriptions contain more stems than the solely resource taxonomic term used in the LCI. As item descriptions are mostly not longer than 10 words, a similarity of 0.1 implies that a match is considered to have at least one of the stems similar to the taxonomy. The k-fold cross validation technique used in the ARM is set by a k value equal to 10. We control the sensitivity of association rules to be used as recommendations by setting the minimal support and minimal confidence for an association rule. These values are initialized with 0.1 as the minimal support and 0.5 as the minimal confidence for a rule to become a recommendation rule.

6.3. Recommender evaluation

The final phase of the experiment involves running the algorithms on the data set and evaluating the performance of the recommenders. Recommender evaluation is usually first performed in an off-line setting using sample data. Such data is either relevant to the context of the recommender or derived from the system for which the recommender is being designed (Ekstrand et al., 2011). Recommenders seek to predict the preference of a user for a certain item. These predictions are estimates of how much the user likes an item and often result in a score represented in an ordinal scale. On the other hand, recommendations are suggestions for items the user might like. Therefore, in the field of information retrieval, recommendations often are evaluated by measuring the prediction effectiveness following a binary classification. In other words, we classify the predictions into recommended items and non-recommended items and use this binary classification to measure the performance of an algorithm. The metrics for evaluating the performance of an recommender algorithms are precision, recall, accuracy and the F-measure, that are defined below.

In a recommender context, the precision and recall measures are described in terms of a set of retrieved items and the set of relevant items to a user. In Equation 1, 2, 3 let tp denote the number of true positives, fp the number of false positives, tn the number of true negatives, and fn the number of false negatives. Precision is the fraction of recommended items that were relevant to a specific user (see Equation 1). Recall is the fraction of relevant recommended items that were retrieved (see Equation 2). Accuracy is the fraction of measurements to be of true value in which the proximity of true value measurements consists of both the recommended items relevant to a user and the non-relevant items that were not recommended (see Equation 3). The F-measure is a different measure of a test's accuracy that evaluates the precision and recall in a weighted harmonic mean (see Equation 4).

$$Precision = \frac{tp}{tp + fp} \tag{1}$$

$$Recall = \frac{tp}{tp + fn} \tag{2}$$

$$Accuracy = \frac{tp + tn}{tp + tn + fp + fn} \tag{3}$$

$$F\text{-}measure = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{4}$$

To make a correct measurement of the relevance of a result, one needs to define the contextual boundaries of what is being measured. Based on our assumption that many items in a resource marketplace are considered by users as similar products from a purpose-based point of view (e.g. recommending iron scrap from one organization is as good as those of other organizations), we evaluate the performance of a recommender algorithm at the level of our defined latent product similarity. Thus, for measuring the effectiveness of an algorithm, we presume that a recommendation is a valid recommendation if a user has expressed an interest in one of the items belonging to the item-cluster. The measure of this type of evaluation is referred to as cluster-level metrics. However, to provide a second perspective on recommender performance it is advisable to perform the measurement also at the level of predicted items. Then, a recommendation is considered valid for an item if the user has expressed interest specifically in that item and not just its cluster. This second type of evaluation is referred to as item-level metrics.

6.4. Algorithm evaluation results

Table 2 shows the results of our experiments. The results are presented with three categorical variables. In the first column we distinguish between the two methods that were applied. In the second column we refer to the level of measurement, as explained in Section 6.3. The third column defines the data set used to test the algorithm. Note that the combined average is constructed by first aggregating the recommendation results of the individual regional experiments in order to calculate the combined performance metrics. Furthermore, we present the percentage of items that were recommended, and the scores achieved on the previously defined performance metrics: precision, recall, accuracy and the F1 or F-measure.

An interesting comparison between the IO-method and the ARM method is performed at the cluster level. The results demonstrate that the ARM method outperforms the IO method on average with a difference ranging from 2 to 3 times in precision, 2.5 to 4 times on accuracy and around 3 times on the F-measure. To a lesser extent we observe a similar pattern when measuring the performance on the item level. At the item level we notice a difference in performance ranging from a factor of 2 to 3. The reader should be aware that predictions at the item level are more prone to be influenced by market characteristics. For example, the number of available items associated with a latent product or the extent to which organizations purchase multiple equivalent products may increase or decrease this number substantially. Another performance metric that requires explanation is accuracy. The results show that the accuracy values achieved in the experiments are somewhat closer to each other, ranging from 0.7 to 0.9. However, accuracy may give a deceptive positive reflection on the performance of the recommender. Although it is true that the accuracy results indicate that many items were correctly classified, the majority of correct classifications here consists of items that were correctly rejected for recommendation. We believe that the effectiveness of a recommender is more clearly characterized by the precision rate along with a reasonable recall, than by the overall accuracy result. Therefore, we include the F-measure, which provides such a combined perspective on precision and recall, and demonstrates the difference in performance of the recommenders better. Furthermore, we analyze the potential regional differences which may have affected the performance of our algorithms. In the application of the IO method no severe differences seem to exist between the regions. However, we observe that the ARM method in Region B performs considerably better at precision than in Region A, while no such precision increase is achieved with the IO method. This can be explained by the size of the data set of region B causing the model to overfit on the rules learned. Therefore, it is reasonable that this higher precision is caused by a few good rules that worked apparently quite well for region B.

Table 2: A comparison of recommender performance

Method	Level	Region	Recom. items	Precision	Recall	Accuracy	F1
IO	Cluster	A	0.1417	0.1210	0.1792	0.7969	0.1444
IO	Cluster	B	0.1717	0.1764	0.2608	0.7727	0.2105
IO	Cluster	A+B	0.1462	0.1309	0.1938	0.7932	0.1562
IO	Item	A	0.1706	0.0564	0.2061	0.8020	0.0885
IO	Item	B	0.2570	0.0516	0.1904	0.7000	0.0812
IO	Item	A+B	0.1832	0.0554	0.2029	0.7871	0.0870
ARM	Cluster	A	0.1986	0.3409	0.7075	0.8411	0.4601
ARM	Cluster	B	0.1616	0.5312	0.7391	0.8939	0.6182
ARM	Cluster	A+B	0.1930	0.3650	0.7131	0.8491	0.4829
ARM	Item	A	0.3089	0.1081	0.7152	0.7111	0.1877
ARM	Item	B	0.2006	0.2727	0.7857	0.8391	0.4049
ARM	Item	A+B	0.2931	0.1245	0.7294	0.7298	0.2126

7. Discussion

7.1. The role of explicit and implicit knowledge

In general, the data suggests a preference to utilize tacit knowledge over explicit knowledge in recommender design. In both regional data sets measured at item and cluster levels, the Association Rule Mining (ARM) algorithm is clearly the better performer in most recommender metrics, except for accuracy that shows a weaker difference and is less consistent. The data shows that the proposed Input-Output (IO) recommender is able to generate recommendations and that in some cases the recommender was able to deduce item preference given the fact that preferred items were retrieved. This demonstrates the value of using explicit knowledge derived from Life Cycle Inventory (LCI) databases in recommendation. However, we consider the precision of the IO algorithm to be too small to be an effective predictor in the context of an industrial symbiotic marketplace. Our findings confirm the ideas of Grant et al. (2010) regarding the challenge of tacit knowledge as one of the characteristics of industrial symbiotic markets that partly determine the success of decision support tools.

The data shows that for a small set of items we are able to predict with higher likelihood that organizations will like these items. This suggests a role for recommenders in IS markets to attract organizations by engaging a critical mass of companies that sustain sufficient supply-demand in the marketplace. The recommenders find application in e.g. newsletters, used as a recurring tool for organizations to be notified of new items after they have participated in workshop sessions. In such a way, slowly we may grow from separate business transactions at workshops towards an active private e-marketplace. The prerequisite for the success of such a recommender is that the precision is large enough to attract user activity that might lead to new investigations of potential symbiotic relations.

7.2. Recommender performance in e-commerce

To take a comparative view on the recommender performance of our experiments with IS items we provide an overview of results achieved in alternative e-commerce systems. Association rule mining is often applied as a recommender technique in different e-commerce domains. However, based on our literature review, only a limited number of studies report on the performance of ARM recommender algorithms. A few studies provide an indication of the precision and recall that can be achieved with the ARM technique. We list the results of those studies which are most similar to our setting where we use a minimal support of 0.1 and a minimal confidence of 0.5, and evaluate each recommendation individually. Thus we do not evaluate the predictions by selecting a top n list of recommendations.

In the study of Mobasher et al. (2001) ARM is used to predict the likelihood that users visit a news article. With a window of 4 transactions, a minimal support of 0.04 and a confidence level of 0.5, they achieve a precision of 0.55 and a recall of 0.45. Lin et al. (2002) propose an adaptive ARM in which the minimal support is controlled during the mining process. They map ratings to transactions in the EachMovie dataset, apply their algorithm and achieve a precision of 0.57 and a recall of 0.76 at a minimal support of 0.1 and a confidence level of 0.9. Swamy and Reddy (2015) present an association rule based recommender method that considers diversity as a controllable aspect. Their algorithm, optimized for High Confidence (HC) and applied on the MovieLens data set appears to be able to achieve a precision of 0.10, but an extremely low recall below 0.01, which might be caused by the choice for a low support of 0.1 in combination with a minimal confidence of 0.05. Another study shows the applicability of ARM in recommending features of products in e-catalogs (Hariri et al., 2013). Their recommendation performance is measured to predict at a precision of 0.54 and a recall of 0.24, based on a confidence level of 0.5 and without any minimal support (Hariri et al., 2013). When considering collaborative-based techniques, Cacheda et al. (2011) show that even rates of precision around 0.85 and with a recall of 0.35 can be reached.

The studies show that prediction accuracy results vary over the different domains of application in e-commerce. A majority of the studies report a moderately better balance between prediction and recall than achieved in our ARM application. Although no firm conclusions may be drawn from such comparison as user behavior, that affects the strength of an algorithm, likely varies a little over the domains. However, it indicates that there may be potential to increase performance with different settings. Moreover, an ongoing discussion exists in recommender system research regarding the optimization of precision, recall or accuracy. The quality of recommendation could be improved also by suggesting a large variety of new and novel items. Furthermore, an IS marketplace can also be characterized by more specialized interest which affects the performance of the ARM technique.

7.3. Data challenges to explicit knowledge-based IS recommenders

Since organizations have become aware of the value of predictive analytics in supply chain management, data quality has been suggested as one of the major challenges that needs to be addressed (Hazen et al., 2014). Explicit knowledge-based recommenders utilize supply chain data to reveal the IS opportunities for organizations. Such methods require data from multiple involved partners potentially enhanced with linkable external data. To understand the problem of data quality in explicit knowledge-based recommendations, we need to address the different dimensions of data quality (e.g. accuracy, timeliness, consistency, and completeness), and the ontological problems that affect the effectiveness of prediction or matchmaking technologies (Cecelja et al., 2015). Recommenders operate with data from industrial symbiotic markets gathered under real world business conditions. As a result, the level of detail and information richness is generally limited, approximate estimates are used in the quantitative values, and often a variety of data formats are used, leading to unstructured data. Poor data quality not only hinders the calculation of an industrial symbiosis match, it also hinders better identification of potential opportunities.

During our design of the IO recommender, a number of data quality problems were encountered. Table 3 illustrates the causes of these knowledge mismatches that result in bad or missing item-recommendations. We classify the data challenges with a severity indication based on the expected frequency of each type of challenge, based on our experience with the IS data. A number of studies illustrate these data challenges to knowledge-based recommenders in order to make more effective predictions. For example, a waste offer that lists certain types of bio-materials that can be used to produce bio-energy may not be directly linked to a demand of bio-energy. In order to let a system suggest a logical knowledge-based recommendation, that system should be aware of the link between bio-materials as resource to produce bio-energy, or should be able to infer the relation between the offered waste and the demand for bio-energy based on e.g. linked or historical data.

Table 3: Data challenges to input-output based industrial symbiosis identification

Data challenge	Severity
- Use of different waste or material taxonomies	High
- Addressing waste at multiple nodes or different levels in the hierarchy of a taxonomy	High
- Limited detail in waste descriptions	High
- Limited structured data available within the public domain on material alternatives to identify substitution	High
- Derive a decomposition of materials from waste descriptions to reveal individual opportunities	Mid
- The used process inputs data may not reflect the actual industry interests	Mid
- Production processes data used to compose the organizational profile do not perfectly reflect the actual production process of that organization	Mid
- Process input data from LCI databases are inaccurate or outdated	Low
- Process data from either organizations or LCI databases can be unreliable (accurate, timely, trustworthy, continuant)	Low
- Process data is limitedly quantitative (related to production frequency, production volumes and waste-quality)	Low

A major problem we experienced concerns the different hierarchical levels in which wastes are addressed. A typical example is the relation between iron and metal. Iron, referring to raw materials largely consisting of the chemical element ferrum, that is a type of metal and would likely be found in scrap metal. However, without such an explicit relation, text-mining algorithms often struggle to relate the two. Cecelja et al. (2015) worked on these challenges of semantic representation for input-output matching with ontological engineering techniques. They show the complexity of modeling explicit knowledge into an IS ontology, but also demonstrate the feasibility of their approach to support matchmaking (Cecelja et al., 2015). The gap to employ these models in practical applications in which industries engage to provide the required data however remains a challenge. Another problem is that some IS can not be identified because data does not provide explicit information on the decomposition of material mixtures which often appear in IS (Yuan et al., 2013). Furthermore, generally no data is available to link waste to the alternatives in order to reveal IS opportunities based on substitution (Hein et al., 2015). On the other hand, data quality is challenged for considerations on the extent to which data attributes can be used within a recommender model. For example, scholars report data problems related to the uncertainty of production volumes (Leong et al., 2016), missing product and transaction information (Dhanorkar et al., 2015), and trustworthiness of data e.g. strategic misleading information (Sheng et al., 2005).

7.4. Limitations and future work

The evaluation provides strong evidence that in a practical IS context, algorithms based on implicit knowledge are better in predicting item preference than those conceptualized from explicit knowledge. Nevertheless, this study has some limitations. First, we selected and constructed two types of algorithms which represent the distinction between explicit and implicit knowledge. A more in depth analysis might include other techniques to strengthen the validity of our research. Furthermore, the LCI database used to create company profiles has its limitation as well. Not all industrial manufacturer's processes could be identified, thereby resulting in fewer recommendations. Another aspect for improvement is to train and test the algorithms on a larger data set. Hence, the ARM is not influenced by potentially negative effects e.g. by overfitting. Finally, raw-material waste items and process industry organizations were selected to study the role of tacit and explicit knowledge in the design of an IS recommender. IS is a much wider concept that goes beyond the exchange of raw materials only. This opens the possibility to design and test algorithms more specifically to recommend other types of IS, such as; waste energy, tools & machinery or over-capacity in a service based industry.

8. Conclusion

This paper analyzes the role of knowledge for the design of waste recommendation algorithms for raw materials. A recurring type of research in Industrial Symbiosis (IS) is the development of software applications to support the process of facilitating symbiotic development. The major problems for the adoption of these tools in practice are the lack of socializability and the data challenges arising from identifying potential matches in the marketplace. The design of recommenders that can effectively suggest opportunities of waste-exchange in IS markets might contribute to the engagement of users in such e-markets.

One of the contributions of our paper is the explicit knowledge based IO recommender. Our proposed recommender algorithm uses manufacturing profiles created with input-output data of life-cycle inventory databases. Such a recommender demonstrates the applicability of knowledge-based recommendations. The other contribution of our paper is in comparing the performance of this explicit knowledge recommender with an implicit or tacit knowledge based recommender. Our results indicate that the performance of the IO recommender is considerably lower than recommender algorithms that rely on tacit knowledge. We argue that such a performance difference can be explained by a number of data challenges that first have to be solved before knowledge-based recommendation can be of practical value to IS decision support tools. On the other hand, we observe a noticeable role for tacit-knowledge based techniques in material recommendation. The performance of tacit knowledge based recommender could enable its acceptance among industries that intend to investigate potential IS opportunities. Finally, the results clearly demonstrate the benefits and limitations of both the IO method and the ARM method, as well as the challenges and preconditions for a successful recommender design in IS markets. Moreover, we confirm that recommender evaluation is essential to the design of an effective prediction algorithm. Future research will focus on the design of recommenders in a broader perspective of the identified IS categories, including waste energy, service and manufacturing tools. Furthermore, testing different recommendation techniques and conducting a sensitivity analysis may contribute to a better understanding and optimization of the current recommender design.

9. Acknowledgements

This research is funded by European Union's Horizon 2020 program under grant agreement No. 680843.

References

Agrawal, R., Imieliński, T., Swami, A., Jun. 1993. Mining association rules between sets of items in large databases. SIGMOD Rec. 22 (2), 207–216. URL `http://doi.acm.org/10.1145/170036.170072`

Agrawal, R., Srikant, R., 1994. Fast algorithms for mining association rules in large databases. In: Proceedings of the 20th International Conference on Very Large Data Bases. VLDB '94. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 487–499.
URL `http://dl.acm.org/citation.cfm?id=645920.672836`

Alfaro, F., Solano, J., 2015. Apriori vs FP-Growth for Frequent Item Set Mining. `http://singularities.com/blog/2015/08/apriori-vs-fpgrowth-for-frequent-item-set-`
, accessed: 2017-01-31.

Boons, F., Chertow, M., Park, J., Spekkink, W., Shi, H., 2016. Industrial symbiosis dynamics and the problem of equivalence: Proposal for a comparative framework. Journal of Industrial Ecology.

Burke, R., 2002. Hybrid recommender systems: Survey and experiments. User Modeling and User-Adapted Interaction 12 (4), 331–370.
URL `http://dx.doi.org/10.1023/A:1021240730564`

Cacheda, F., Carneiro, V., Fernández, D., Formoso, V., Feb. 2011. Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems. ACM

Trans. Web 5 (1), 2:1–2:33.
URL http://doi.acm.org/10.1145/1921591.1921593

Cecelja, F., Raafat, T., Trokanas, N., Innes, S., Smith, M., Yang, A., Zorgios, Y., Korkofygas, A., Kokossis, A., 2015. e-symbiosis: technology-enabled support for industrial symbiosis targeting small and medium enterprises and innovation. Journal of Cleaner Production 98, 336 – 352, special Volume: Support your future today! Turn environmental challenges into opportunities.
URL //www.sciencedirect.com/science/article/pii/S0959652614008749

Chen, Y., Canny, J. F., 2011. Recommending ephemeral items at web scale. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '11. ACM, New York, NY, USA, pp. 1013–1022.
URL http://doi.acm.org/10.1145/2009916.2010051

Chen, Z., Li, H., Kong, S. C., Hong, J., Xu, Q., 2006. E-commerce system simulation for construction and demolition waste exchange. Automation in Construction 15 (6), 706 – 718, knowledge Enabled Information System Applications in Construction.
URL //www.sciencedirect.com/science/article/pii/S0926580505001287

Chertow, M., Ehrenfeld, J., 2012. Organizing self-organizing systems. Journal of Industrial Ecology 16 (1), 13–27.
URL http://dx.doi.org/10.1111/j.1530-9290.2011.00450.x

Chertow, M. R., 2000. Industrial symbiosis: literature and taxonomy. Annual review of energy and the environment 25 (1), 313–337.

Chertow, M. R., 2007. uncovering industrial symbiosis. Journal of Industrial Ecology 11 (1), 11–30.
URL http://dx.doi.org/10.1162/jiec.2007.1110

Clayton, A., Muirhead, J., Reichgelt, H., 2002. Enabling industrial symbiosis through a web-based waste exchange. Greener Management International 40, 93–107.

Cutaia, L., Luciano, A., Barberio, G., Sbaffoni, S., Mancuso, E., Scagliarino, C., La Monica, M., 2015. The experience of the first industrial symbiosis platform in italy. Environmental Engineering & Management Journal 14 (7), 1521–1533.

Dhanorkar, S., Donohue, K., Linderman, K., 2015. Repurposing materials and waste through online exchanges: overcoming the last hurdle. Production and Operations Management 24 (9), 1473–1493.

Dietrich, J., Becker, F., Nittka, T., Wabbels, M., Modoran, D., Kast, G., Williams, I., Curran, A., den Boer, E., Kopacek, B., et al., 2014. Extending product lifetimes: a reuse network for ict hardware. Proceedings of the ICE-Waste and Resource Management 167 (WR3), 123–135.

Ecoinvent, 2017. The ecoinvent database (version 3.3). http://www.ecoinvent.org/database/database.html, accessed: 2017-02-01.

Ekstrand, M. D., Riedl, J. T., Konstan, J. A., Feb. 2011. Collaborative filtering recommender systems. Found. Trends Hum.-Comput. Interact. 4 (2), 81–173.
URL http://dx.doi.org/10.1561/1100000009

European Commission, 05 2000. Commission decision on the european list of waste (com 2000/532/ec). Tech. rep., European Commission.

European Environmental Agency, 10 2016. More from less material resource efficiency in europe. eea report no 10/2016. Tech. rep., European Environmental Agency.

Freyne, J., Jacovi, M., Guy, I., Geyer, W., 2009. Increasing engagement through early recommender intervention. In: Proceedings of the Third ACM Conference on Recommender Systems. RecSys '09. ACM, New York, NY, USA, pp. 85–92.
URL http://doi.acm.org/10.1145/1639714.1639730

Gibbs, D., Deutz, P., 2007. Reflections on implementing industrial ecology through eco-industrial park development. Journal of Cleaner Production 15 (17), 1683 – 1695, from Material Flow Analysis to Material Flow Management.
URL http://www.sciencedirect.com/science/article/pii/S095965260700039X

Gomez-Uribe, C. A., Hunt, N., Dec. 2015. The netflix recommender system: Algorithms, business value, and innovation. ACM Trans. Manage. Inf. Syst. 6 (4), 13:1–13:19.
URL `http://doi.acm.org/10.1145/2843948`

Grant, G. B., Seager, T. P., Massard, G., Nies, L., 2010. Information and communication technology for industrial symbiosis. Journal of Industrial Ecology 14 (5), 740–753.
URL `http://dx.doi.org/10.1111/j.1530-9290.2010.00273.x`

Han, J., Pei, J., Yin, Y., May 2000. Mining frequent patterns without candidate generation. SIGMOD Rec. 29 (2), 1–12.
URL `http://doi.acm.org/10.1145/335191.335372`

Hariri, N., Castro-Herrera, C., Mirakhorli, M., Cleland-Huang, J., Mobasher, B., Dec. 2013. Supporting domain analysis through mining and recommending features from online product listings. IEEE Trans. Softw. Eng. 39 (12), 1736–1752.
URL `http://dx.doi.org/10.1109/TSE.2013.39`

Hazen, B. T., Boone, C. A., Ezell, J. D., Jones-Farmer, L. A., 2014. Data quality for data science, predictive analytics, and big data in supply chain management: An introduction to the problem and suggestions for research and applications. International Journal of Production Economics 154, 72 – 80.
URL `//www.sciencedirect.com/science/article/pii/S0925527314001339`

Hein, A. M., Jankovic, M., Farel, R., Sam, L. I., Yannou, B., 2015. Modeling industrial symbiosis using design structure matrices. In: 17th international dependency and structure modeling conference, DSM 2015.

International Synergies Ltd., 2016. Training on industrial symbiosis taxonomies. Personal communication.

Jung, Y., Park, H., Du, D.-Z., Drake, B. L., 2003. A decision criterion for the optimal number of clusters in hierarchical clustering. Journal of Global Optimization 25 (1), 91–111.
URL `http://dx.doi.org/10.1023/A:1021394316112`

Leong, Y. T., Tan, R. R., Aviso, K. B., Chew, I. M. L., 2016. Fuzzy analytic hierarchy process and targeting for inter-plant chilled and cooling water network synthesis. Journal of Cleaner Production 110, 40–53.

Lin, W., Alvarez, S. A., Ruiz, C., 2002. Efficient adaptive-support association rule mining for recommender systems. Data Mining and Knowledge Discovery 6 (1), 83–105.
URL `http://dx.doi.org/10.1023/A:1013284820704`

Lombardi, D. R., Laybourn, P., 2012. Redefining industrial symbiosis. Journal of Industrial Ecology 16 (1), 28–37.
URL `http://dx.doi.org/10.1111/j.1530-9290.2011.00444.x`

Manning, C. D., Raghavan, P., Schütze, H., et al., 2008. Introduction to information retrieval. Vol. 1. Cambridge university press Cambridge.

Mirata, M., 2004. Experiences from early stages of a national industrial symbiosis programme in the uk: determinants and coordination challenges. Journal of Cleaner Production 12 (810), 967 – 983, applications of Industrial Ecology.
URL `http://www.sciencedirect.com/science/article/pii/S0959652604000848`

Mobasher, B., Dai, H., Luo, T., Nakagawa, M., 2001. Effective personalization based on association rule discovery from web usage data. In: Proceedings of the 3rd International Workshop on Web Information and Data Management. WIDM '01. ACM, New York, NY, USA, pp. 9–15.
URL `http://doi.acm.org/10.1145/502932.502935`

Natural Language Toolkit, 2017. Nltk 3.0 documentation. `http://www.nltk.org/`, accessed: 2017-01-31.

Natural Resources Wales, Scottish Environment Protection Agency, Northern Ireland Environment Agency, Environment Agency, 2015. Waste classification: Guidance on the classification and assessment of waste (1st edition 2015). Tech. rep.

Paquin, R. L., Howard-Grenville, J., 2012. The evolution of facilitated industrial symbiosis. Journal of Industrial Ecology 16 (1), 83–93.
URL `http://dx.doi.org/10.1111/j.1530-9290.2011.00437.x`

Park, D. H., Kim, H. K., Choi, I. Y., Kim, J. K., 2012. A literature review and classification of recommender systems research. Expert Systems with Applications 39 (11), 10059 – 10072.
URL //www.sciencedirect.com/science/article/pii/S0957417412002825

Pathak, B., Garfinkel, R., Gopal, R., Venkatesan, R., Yin, F., Oct. 2010. Empirical analysis of the impact of recommender systems on sales. J. Manage. Inf. Syst. 27 (2), 159–188.
URL http://dx.doi.org/10.2753/MIS0742-1222270205

Peffers, K., Tuunanen, T., Rothenberger, M. A., Chatterjee, S., 2007. A design science research methodology for information systems research. Journal of Management Information Systems 24 (3), 45–77.
URL http://www.tandfonline.com/doi/abs/10.2753/MIS0742-1222240302

Porter, M. F., 1980. An algorithm for suffix stripping. Program 14 (3), 130–137.

Salvador, S., Chan, P., 2004. Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. In: Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence. ICTAI '04. IEEE Computer Society, Washington, DC, USA, pp. 576–584.
URL http://dx.doi.org/10.1109/ICTAI.2004.50

Sander, K., Schilling, S., Lskow, H., Gonser, J., Schwedtje, A., Kchen, V., 11 2008. Review of the european list of waste. Tech. rep., kopol GmbH and ARGUS GmbH.

Sheng, Y. P., Mykytyn Jr, P. P., Litecky, C. R., 2005. Competitor analysis and its defenses in the e-marketplace. Communications of the ACM 48 (8), 107–112.

Sterr, T., Ott, T., 2004. The industrial region as a promising unit for eco-industrial developmentreflections, practical experience and establishment of innovative instruments to support industrial ecology. Journal of Cleaner Production 12 (810), 947 – 965, applications of Industrial Ecology.
URL http://www.sciencedirect.com/science/article/pii/S0959652604000836

Swamy, K. M., Reddy, K. P., 2015. Improving Diversity Performance of Association Rule Based Recommender Systems. Springer International Pub-

lishing, Cham, pp. 499–508.
URL http://dx.doi.org/10.1007/978-3-319-22849-5_34

The ISDATA project, 2015. The industrial symbiosis data repository. http://isdata.org/, accessed: 2017-01-31.

United Nations Statistics Division, 2015. Central product classification (version 2.1). http://unstats.un.org/unsd/cr/registry/cpc-21.asp, accessed: 2017-01-31.

Van Beers, D., Corder, G., Bossilkov, A., Van Berkel, R., 2007. Industrial symbiosis in the australian minerals industry. Journal of Industrial Ecology 11 (1), 55–72.

Veiga, L. B. E., Magrini, A., 2009. Eco-industrial park development in rio de janeiro, brazil: a tool for sustainable development. Journal of Cleaner Production 17 (7), 653 – 661, present and Anticipated Demands for Natural Resources: Scientific, Technological, Political, Economic and Ethical Approaches for Sustainable Management.
URL //www.sciencedirect.com/science/article/pii/S095965260800293X

Wroblewska, A., Twardowski, B., Zawistowski, P., Ryżko, D., 2016. Automatic Clustering Methods of Offers in an E-Commerce Marketplace. Springer International Publishing, Cham, pp. 147–160.
URL http://dx.doi.org/10.1007/978-3-319-30315-4_13

Yuan, X., Lee, J.-H., Kim, S.-J., Kim, Y.-H., 2013. Toward a user-oriented recommendation system for real estate websites. Information Systems 38 (2), 231 – 243.
URL http://www.sciencedirect.com/science/article/pii/S0306437912001081

Zheng, Z., Kohavi, R., Mason, L., 2001. Real world performance of association rule algorithms. In: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '01. ACM, New York, NY, USA, pp. 401–406.
URL http://doi.acm.org/10.1145/502512.502572

Appendix A. Appendix

Appendix A.1. Stem frequency vectorization

Algorithm 3: Stem-frequency vectorization of waste item-descriptions

```
Data:
        $N_{item}$ = Marketplace item to be vectorized
        $W_{items}$ = Set of all waste items in marketplace
        U = Set of unique stems
Result:
         V = Stem-frequency item vector
Function Stem-Frequency-Vectorization($N_{item}$, $W_{items}$)
    Create a bag of stems $N_{bag}$ for item $N_{item}$;
    Create a bag of stems $W_{bags}$ for each item in $W_{items}$;
    foreach $bag \in W_{bags}$ do
        foreach $stem \in bag$ do
            if $stem \notin U$ then
                Add the unique $stem$ to $U$;
    Initialize an empty item vector $V$ with $n$ positions where $n$ is equal to
      the number of unique stems $U$;
    foreach stem $s \in N_{bag}$ do
        Find the position $p$ of stem $s$ in the set of unique stems $U$;
        Increase the frequency with $+1$ in the item vector $V$ at position $p$ ;
    return $V$;
```

Algorithm 4: A simple multi-dimensional hierarchical agglomerative clustering algorithm

```
Data:
        $W_{items}$ = Set of all waste items in marketplace
        $S$ = Matrix of cosine similarities
        $V$ = Stem-frequency item vector
        $p_1$ = Maximum number of iterations
        $p_2$ = Minimum similarity cut-off criterium;
Result:
        C = List of clusters
Function Simple-MHAC($W_{items}$, $p_1$, $p_2$)
    foreach $item \in W_{items}$ do
        Create a stem-frequency vector $V_{item}$ and store in $V$;
    for $m \leftarrow 1$ to $W_{items}$ do
        for $n \leftarrow 1$ to $W_{items}$ do
            if $n > m$ then
                Calculate the cosine similarity $s$ for item combination
                 $\{n,m\}$ using the associated item vectors $V_n$, and $V_m$ and
                 store the similarity $s_{\{m,n\}}$ in the similarity matrix $S$;
        Assign item $n$ to its own cluster $c_{\{n\}}$ in store cluster in $C$;
    for $iteration \leftarrow 1$ to $W_{items} - 1$ do
        Determine the key pair $\{m, n\}$ of the maximal similarity $s_{max}$ in
         the matrix S.;
        if $iteration <= p_1$ and $s_{max} >= p_2$ then
            Merge cluster $C_m$ and $C_n$ into a new cluster $C_{\{m,n\}}$;
            Remove cluster $C_m$ and $C_n$ from $C$;
            Remove similarities $s_m$ and $s_n$ from $S$;
            Merge the item vector $V_m$ and $V_n$ into a new cluster vector
             $V_{\{m,n\}}$;
            foreach $cluster \in C$ do
                Calculate the cosine similarity $s$ for item combination
                 $\{n,m\}$ and $cluster$ using the associated item vectors
                 $V_{\{n,m\}}$, and $V_{cluster}$ and store the new similarity
                 $s_{\{\{m,m\},cluster\}}$ in the similarity matrix $S$;
    return $C$;
```