Connected Closet
A Semantically Enriched Mobile Recommender System for Smart Closets

Anders Kolstad[1], Özlem Özgöbek[1], Jon Atle Gulla[1] and Simon Litlehamar[2]
[1]*Department of Computer and Information Science, NTNU, Trondheim, Norway*
[2]*Accenture AS, Fornebu, Norway*

Keywords: Recommender Systems, Web of Data, Semantic Web, Mobile Systems, Fashion Recommendation, Internet of Things.

Abstract: A common problem for many people is deciding on an outfit from a vastly overloaded wardrobe. In this paper, we present Connected Closet, a semantically enriched Internet of Things solution of a smart closet with a corresponding mobile application for recommending daily outfits and suggesting garments for recycling or donation. This paper describes the whole design and architecture for the system, including the physical closet, the recommender algorithms, the mobile application, and the backend comprising of microservices implemented using container technology. We show how users can benefit from the system by supporting them in organizing their wardrobe, and receiving daily personalized outfit suggestions. Moreover, with the system's recycling suggestions, the system can be beneficial for the sustainability of the environment and the economy.

1 INTRODUCTION

According to a technical report by the National Institute for Consumer Research, the typical Norwegian has on average 80.4 kg of clothes in their closet (Klepp and Laitala, 2016). This is equivalent to 359 different clothing items. This suggests that Norwegians have a need for better structuring of the clothing items in their closets. Moreover, the report states that 20% of the clothes were never used or only used a couple of times. This might suggest that the person did not actually like the item they bought from the clothing retailer. Today, many organizations and clothing retailers offer checkpoints where people can deliver garments for recycling or donation. This benefits the sustainability for both the environment and the economy (Chavan RB, 2014). Moreover, Pruit (2015) argues that our selection of an outfit influences others' impressions of us and that careful selection of an outfit is of high importance to our social and cultural lives.

It has long been spoken of the huge amount of data generated by user-generated content on the Web. Because of this exponential growth of data, the era of big data has arised (Jagadish et al., 2014). This huge growth of data has resulted in information overload. This implies a clear need for applications able to help users navigate through the vast amount of content in a personalized way. Such applications can be made a reality by recommender systems. Furthermore, this vast amount of data also calls for a need to structure the data available online in a meaningful way (Bizer et al., 2009). As a result, a huge amount of Resource Description Framework (RDF) data has been published as Linked Open Data (LOD) through the Linking Open Data project[1]. This kind of data have huge potential power and recommender systems can benefit from this data, generating even more accurate and personalized recommendations (Figueroa et al., 2015).

In this paper, we present Connected Closet, a semantically enriched Internet of Things (IoT) smart closet, using clothing items enabled with radio-frequency identification (RFID) tags, keeping track of the clothing items' usage history. We describe an implementation of a mobile application prototype and its backend where the users can keep track of what items currently are in their closets, get recommendations on what to wear for today, and get recommendations on what to donate and recycle. Furthermore, we propose three different recommender approaches in the domain of fashion recommendation.

The work presented in this paper is a collaborative

[1]https://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData

Kolstad, A., Özgöbek, Ö., Gulla, J. and Litlehamar, S.
Connected Closet - A Semantically Enriched Mobile Recommender System for Smart Closets.
DOI: 10.5220/0006298002980305
In *Proceedings of the 13th International Conference on Web Information Systems and Technologies (WEBIST 2017)*, pages 298-305
ISBN: 978-989-758-246-2

effort between the Smartmedia program[2] at the Norwegian University of Science and Technology[3] and Accenture Norway[4]. The Smartmedia program focuses on mobile context-aware recommender systems research. The goal of this program is to present a context-aware, personalized news reading experience based on deep understanding of the textual content of news articles. Accenture is a leading management, strategy, consulting, and technology company employing approximately 384,000 people. Using technology, Accenture strives to help businesses create IT applications to deal with rapid changes caused by an increasingly digital economy. Using their *new IT strategy*, they deliver liquid, connected, and intelligent applications to their clients. In this research project, Accenture demonstrate how to realize the new IT strategy, and explore applications in their technology vision by showcasing the advantages of intelligent automation to their clients.

The contributions of this paper are: (1) the architecture and design of a smart IoT closet; and (2) three recommendation techniques for recommending items in the fashion domain.

The rest of the paper is structured as follows. Section 2 introduces the background theory. In Section 3, we give an overview of related work. In Section 4, we describe the smart closet and the user interface for the mobile application, followed by a more detailed description on how the closet is supported by recommendation technology in Section 5. We conclude with a summary, and discuss future work and possible benefits of the system.

2 BACKGROUND

2.1 Linked Data

Today, most of the data on the Web is uploaded as HTML documents, or raw dumps such as CSV. This way of uploading data sacrifices much of the Web's structure and semantics. Bizer et al. (2009) has outlined a set of rules for publishing data by using the Web to create typed links between the data. By publishing data according to these rules, computer agents can read and make sense of the data published, making it easier to gather meaningful information from the Web. Linked Data published under an open license is called LOD. An example of LOD is Wikidata[5], which is an open knowledge base that can be read or edited by any human or computer agent. Using Wikipedia, Wikivoyage, Wikisource, and others as its central storage, Wikidata contains millions of RDF triples, following a Subject, Predicate, Object structure exposed on its SPARQL endpoint.

2.2 Recommender Systems

The objective of a recommender system is to guide users in making choices by giving them personalized recommendations of items. Typically, recommender systems are classified into collaborative filtering and content-based (Jannach et al., 2010). Collaborative filtering generate recommendations on the idea that if some users shared the same interest on previous items, they will have similar preferences to other items as well. Content-based recommendations base its recommendations on item descriptions and a user profile. A user profile is a set of a user's preferences; the recommender system will then recommend the items that have the most similar item description to the user profile.

An interesting challenge for all recommender systems is computing accurate recommendations when few user ratings are available. This challenge is known as the sparsity problem (Jannach et al., 2010). A special case of the sparsity problem is: (a) dealing with new users who have not yet rated any items; and (b) how to recommend new items that has not been rated yet. These two problems are commonly known as the new-user and new-item cold-start problem.

The majority of today's recommender systems addresses recommendations of items in the domains of movies, books, and music. Different techniques of the approaches above are well researched and evaluated in these domains (Bobadilla et al., 2013).

2.3 Internet of Things

IoT is a set of Internet-connected devices embedded with hardware, software, sensor, actuators, identifiers, and network technologies. These devices collect and exchange data with each other and other components on the Internet, generating a vast amount of data every day (Gubbi et al., 2013). IoT meets the need for data-on-demand by intuitive interactions with ubiquitous computing devices. Furthermore, IoT has been identified as one of the key trends that organizations must keep track of to gain competitive advantage, and that the market adaptation is predicted to take 5–10 years.

[2]http://research.idi.ntnu.no/SmartMedia/
[3]http://www.ntnu.edu/
[4]https://www.accenture.com/no-en/
[5]https://www.wikidata.org/

The applications in the domains that will be and has been impacted by IoT devices range from control of home equipment such as refrigerators, to monitoring the water quality in cities.

3 RELATED WORK

Several works have been done on enabling Linked Data into recommender systems in order to improve their recommendation algorithms (Figueroa et al., 2015).

Heitmann and Hayes (2010) describe a recommender system that tries to mitigate i) the new-item problem; ii) the new user problem; and iii) the sparsity problem of recommendations of music by utilizing Linked Data. They transformed RDF graphs into a user-item matrix by using data from MySpace and data about a Wikipedia editor's homepage. Their results showed that by enabling Linked Data, the average precision increased by 14% and the average recall increased by 33%.

Di Noia et al. (2012) propose a recommender system that relies exclusively on information extracted from the Web of Data. For recommending movies they propose a content-based recommender system using the SPARQL endpoints exposed by DBpedia, LinkedMDB, and Freebase as the base of their recommender system. To compute similarities between movies they used the Vector Space Model, representing the whole RDF as a 3-dimensional matrix where each slice refers to an ontology property. Given a property, each movie is seen as a vector. For a given slice, the similarity is computed between the correlating movie vectors by calculating the cosine angle between the vectors.

Tomeo et al. (2016) generated a dataset consisting of Facebook likes of music, books, and movies. First, they mapped the likes to entities in DBpedia to enrich the item profiles in the dataset. Then, they compared the dataset on different graph-based recommender systems and matrix factorization systems. Overall, the graph-based algorithm, PathRank showed the most promising results.

Many prototypes of smart IoT closets for suggesting outfits by using RFID technology have been made in the past (Goh et al., 2011; Ling et al., 2007; Toney et al., 2006). These prototypes show some very promising techniques for IoT closets and are built on the same fundamental techniques as described in this paper. These techniques involves attaching RFID tags to the clothing items or hangers, which can be scanned by a reader in the closet, and then broadcasting a message about the state of the item to a database.

Moreover, similar to our system, some of them also enable weather data or calendar integration (Schaad et al., 2016; Liu et al., 2012). As far as we know, none of them utilize LOD as we do in our system. Moreover, none of them focus on recommending clothing items that the user might want to donate or recycle. Research and evaluation of the recommender algorithms in these studies are lacking or has been scoped out.

Ingvaldsen et al. (2015) propose a recommendation technique for how personalized and location aware news recommendations can be constructed based on the users' contexts. Moreover, they show how the recommended content can be enriched by using Wikidata. In our prototype of Connected Closet, we use similar techniques to combine Wikidata with context aware user ratings to construct location aware recommendations based on the weather at the user's location.

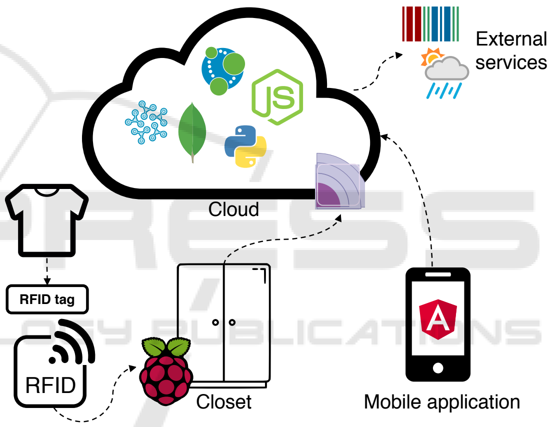


Figure 1: High level architecture.

4 ARCHITECTURE

The main parts of the prototype are constructed as follows. As shown in Figure 1, the closet is embedded with a Raspberry PI[6], a tiny computer, which is connected to an RFID reader. When an end-user touches the RFID tag of a clothing item onto the RFID reader, the Raspberry PI will broadcast a message to the backend of the prototype which is constructed of microservices running in the cloud, each performing their own designated task. The mobile application communicates with the microservices, providing the end-user with recommendations and inventory overview. Moreover, the high level architecture contains external services which consists of third party APIs, such as weather data and LOD.

[6] https://www.raspberrypi.org/

The components of Connected Closet are connected as follows.

4.1 Closet

The computer embedded in the physical closet runs a Python script listening to scans of clothing items. When a scanning occurs, the script broadcasts a Message Queuing Telemetry Transport (MQTT) message containing timestamp, item id, user id of the closet owner, and the status of the clothing item (whether it is being checked in or out of the closet). Additionally, by using LED lights and speakers connected to the computer, the user receives feedback on an item scan. A red light indicates an insertion, while green light indicates extraction. This is implemented to maintain consistency between the physical clothing items in the closet and the status of the items stored by the microservices. To implement this, we used ideas from Fog Computing (Bonomi et al., 2012) and implemented a local cache database in the embedded computer that keeps track of the status of the latest item scans. Additionally, this prevents the Python script in broadcasting unnecessary messages, such as double scans.

Figure 2: Prototype of the physical closet.

Figure 2 shows a picture of an early version of the prototype, including the physical closet embedded with the Raspberry PI. The prototype is built for demonstration purposes, and to show how human interaction with the closet would work in practice.

4.2 Backend

The backend of the prototype (Figure 3) consists of five main components implemented as microservices: the inventory service, the history service, the catalog service, the recommender service, and the closet service. All the microservices and their databases have been implemented using container-based virtualization with Docker[7]. This type of virtualization is an operating-system-level virtualization method for running distributed applications without the need for launching an entire virtual machine. With such container-based microservices, the whole solution benefits from a horizontal scalable architecture composed of small, indepenedent, and highly coupled components communicating with each other by means of Representational State Transfer (REST) with the Hypertext Transfer Protocol (HTTP). The main components of the backend are described throughout the section.

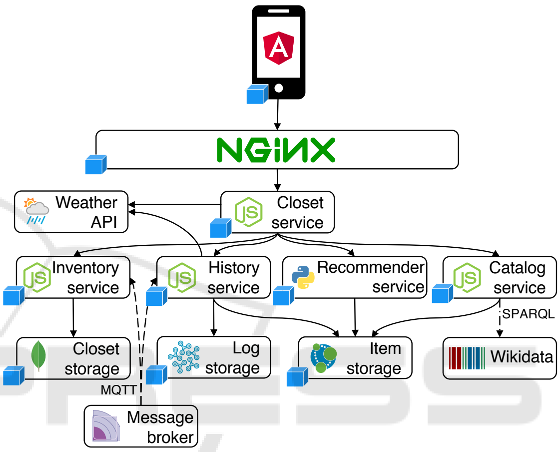


Figure 3: Detailed architecture of the backend.

4.2.1 The Inventory Service

The inventory service is responsible for storing data about the clothing items registered to an owner of a closet. The service stores its data in a document-oriented database. Each owner of a closet is assigned their own document in this database. The owner's clothing items are represented as a list of triples containing the unique id of the clothing item, the article number of the item, and the status of the item (inside or outside of closet). Moreover, this service stores user information such as username and favorite outfits. The favorite outfits are represented as a list of tuples containing two items, one top and one bottom.

When the service receives the MQTT message, the service will then take note of which user the scanning came from and update the user's document in the database.

[7] https://www.docker.com/

4.2.2 The History Service

The history service is responsible for logging every scan that occurs in the closet. A time-series database is connected to this service for storing each log entry. For each scanning, a record containing the item id, the timestamp, status of the item, and current temperature will be saved. Furthermore, this service writes usage history and temperatures to a database shared with the recommender and the catalog service.

4.2.3 The Catalog Service

The catalog service is responsible for handling data about all the different clothing items that are supported by a Connected Closet. A supported item is an item connected to an RFID tag with its article number stored in the database shared with the history and the recommender service. This service handles matching clothing items and other item properties, such as color.

Furthermore, this service is set up to communicate with Wikidata via Wikidata's SPARQL endpoint, semantically enriching the clothing items in the shared database. E.g., if one clothing items is registered to have the color 'navy' and another item is 'blue', the results from Wikidata will include similar description to both of these colors, making the similarity between the item descriptions even stronger. This is also done on other item properties where this is expedient.

4.2.4 The Recommender Service

The recommender service lies in the heart of the recommendation approaches explained in Section 5. This service uses the item ratings and the descriptive item data stored in the shared database, called Item storage. This database is implemented as a graph database. Figure 4 shows a simplified example of how the data is represented in the database.

To realize the recommendation approaches, the service employs different machine learning libraries.

4.2.5 The Closet Service

The closet service is responsible for providing the mobile application with meaningful information gathered from the lower level services. It generates a set of the closet overview by joining the data from the catalog service and the inventory service. Using weather data and item status from inventory as input to the recommender service, the recommender service will return a list of recommended outfits. For getting recycling recommendations from the recommender services, it uses data gathered from the history service.

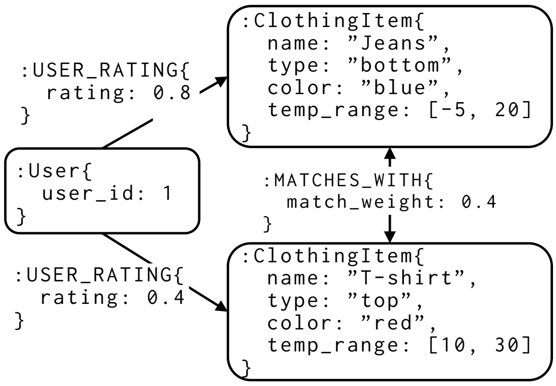


Figure 4: Example of data representation in Item storage.

4.3 Mobile Application

A progressive web application is developed to make the closet explorable on mobile devices. In this application, the user is allowed to view suggestions for today's outfit, view an inventory of their closet, and view suggestions on what clothing items to recycle or donate.

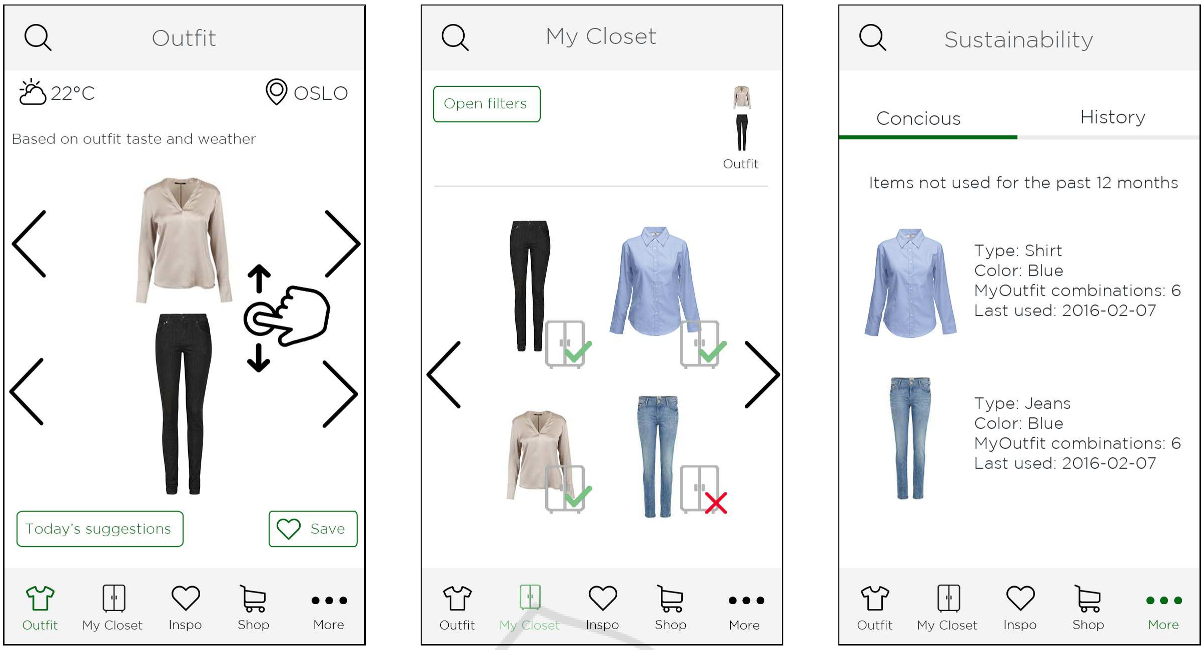
The mobile application communicates with the closet service by REST through HTTP. A web server is set up in the middle to handle traffic and connections using Nginx[8].

4.3.1 User Interface

To view suggestions for today's outfit, the user chooses the Outfit button from the lower menu bar. Figure 5a shows an illustration of the outfit suggestion view. In the top of the view, the weather and location for the user is displayed. Below is the suggested outfit. The user can modify the suggested outfit by clicking on the arrows next to the clothing items of the outfit. If the user wants to go back to what the system has recommended for today, they can use the Today's suggestion button, loading the initial recommendation. The user can save the outfit displayed as a favorite by using the button next to Today's suggestion. Furthermore, the user can browse through a list of top-k outfit recommendations by swiping up and down on the screen.

Figure 5b shows an example of a closet overview. By choosing the My Closet button from the menu, the overview of the user's closet is displayed. Here, the user can browse all the clothing items registered to their closet and see item status indicated by a closet icon with a check mark. Moreover, a filtering function

[8] https://nginx.org/en/



(a) Outfit suggestion (b) Closet overview (c) Recycling suggestion

Figure 5: Screenshots from the Connected Closet prototype.

is implemented to help the user navigate through the closet overview more easily.

By choosing the More button from the menu, the user is displayed with a list of items rarely used and a suggestion to recycle these items. Figure 5c shows a suggestion for two items that have not been used for the past 6 months.

5 FASHION RECOMMENDATION

In this section we describe how the whole system is supported by recommender technology. We describe how the item properties needed to do fashion recommendation are determined. Furthermore, we divide fashion recommendation into outfit recommendation and recycling recommendation.

5.1 Ratings

The user ratings used in fashion recommendation are determined depending on several context factors. The context factors for determining the user rating of an item are:

- Usage history: How often an item has been extracted from the user's closet affects the item's rating. It is safe to assume that a frequently used item is an item that the user likes. Therefore, an item that is used on a weekly basis will have a high rating.
- Current season: Some clothing items are seasonal. E.g., a winter jacket will have higher rating during the winter, and a low rating in the summer.
- Weather: Much like season, some items are weather dependent, e.g., rain coat. The rating of such items will therefore be affected by the daily weather.
- Taste profile: As the user saves outfits as favorites, the items in the outfits will increase their user ratings.

Using these factors as input, the rating of an item is set using a 10-star rating scale.

5.2 Matches and Suitable Temperatures

For determining matching clothing tops and bottoms in the catalog service, we calculate a weight between the two items and assign it to their edge. Initially, all tops and bottoms match each other with a weight of 0.0. When two items are checked out of the closet during the same 2-hour time period, the weight between these items increase with α. Furthermore, when a user saves an outfit as a favorite using the mobile application, the weight increases by β. The matching weight cannot exceed 1.0.

All clothing items are saved in the catalog service with a suitable temperature range property. The suitable temperature range is the range of temperatures in which a clothing item is comfortable to wear. This range is determined by the average temperature of all the checkouts of a clothing item, calculated by Formula 1:

$$st(i) = \frac{1}{N}(\sum_{j=1}^{N} C_{i,j,temp}) \pm \delta\,^{\circ}\mathrm{C}, \quad (1)$$

where N is the number of checkouts of clothing item i in C_i and δ is a constant determining the length of the range.

5.3 Outfit Recommendation

In our system, two approaches for outfit recommendation are implemented. The first approach uses ideas from collaborative filtering, while the second approach is a pure content-based approach enabled with LOD.

For an item to be included in a recommended outfit it must be: (1) inside the closet; and (2) the current temperature must be inside the items suitable temperature range.

5.3.1 Outfit-item Matrix

In the first approach, we transform the outfits saved as favorites by the end-users into an outfit-item matrix. In this matrix, each column represents a favorite outfit of an end-user. The rows represent every item supported by Connected Closet and that is part of a user's favorite outfit. Table 1 shows an example of an outfit-item matrix with three outfits and four items. All outfits is associated with a weight, e.g, w_1. These weights are based on number of likes of the outfit and are used to determine the strength of the outfit, making it easier to neglect outfits favorited by few users. Using this matrix as training data, different classification algorithms are applied to classify outfits as *good* or *neutral*. A *good* outfit means an outfit that can be recommended to the user. While outfits classified as *neutral* are outfits that the users either does not like or has not been rated yet, and will therefore not be recommended to the users. In our method, we first create outfits combinations of the items that fit our inclusion criteria. Then, we input these outfits into the classification model. Outfits that are classified as *good* will then be recommended to the user.

Table 1: Example of an outfit-item matrix.

	Outfit 1 w_1	Outfit 2 w_2	Outfit 3 w_3
Item 1	×	×	
Item 2		×	
Item 3	×		×
Item 4			×

5.3.2 Vector Space Model

For our content-based approach, we use a vector space model similar to the one proposed by Di Noia et al. (2012). Using the user ratings stored in the item storage, we build a user profile consisting of clothing items with a rating above λ, using Formula 2:

$$profile(u) = \{c_i \mid r_{ui} > \lambda\}, \quad (2)$$

where r_{ui} is rating of clothing item c_i for user u.

We then generate a ranked list of all the clothing items in the user's closet using Formula 3:

$$\overline{r}(u, c_i) = \frac{\sum_{c_j \in profile(u)} sim(c_j, c_i)}{|profile(u)|}, \quad (3)$$

where $sim(c_j, c_i)$ is a similarity measure between the vectors representing the clothing items c_j and c_i.

We then filter out a list of top-k outfits based on our inclusion criteria and the match weight between the tops and bottoms.

5.4 Recycling Recommendation

For recommending items that may be of interest for recycling by the end-user, the system returns a list of the three lowest rated items that have been rarely used in the past 12 months. By using a time period of minimum 12 months it safe to recommend items that are also seasonal.

An identified problem with the proposed technique is the new user cold-start problem. The owner of a Connected Closet should be able to get relevant recycling recommendations from the day that they acquire the closet. This problem, and other techniques for recycling recommendation will be explored in later research.

6 CONCLUSION AND FUTURE WORK

The described prototype is an ongoing project with some development still remaining. A full evaluation and validation will be performed in later research. In this paper we have proposed a novel IoT system for doing fashion recommendation using modern technologies, such as LOD, microservices, containers, and progressive web apps. The proposed system can guide users in making daily outfit selections and efficiently organize their wardrobe in an environment-friendly way. Furthermore, if the system were to be integrated in a clothing retailer's supply chain and used by their customers, the retailer could generate targeted ads and provide relevant recommendations to their customers.

As an initial evaluation, the prototype has been showcased at various IT conferences and events. At these events, the participants have been given a demonstration of the prototype and have had the opportunity to try out the prototype for themselves. The response from the participants has been positive, and many participants have expressed that they would benefit from such a system in their everyday lives.

Future work will be devoted to gathering data for a dataset that can be used for accuracy evaluation of the recommendation approaches. Moreover, we aim to do an user-centric evaluation of the recommender system in order to evaluate the user satisfaction. Some other topics we would like to research further and include in our system are occasion-based outfit recommendations and recommendations from retailers.

ACKNOWLEDGEMENTS

This work is composed by a research cooperation that was established subsequently of a summer internship at Accenture, where the idea and the first prototype of Connected Closet originally was developed. The authors would like to thank everyone that was involved in the internship for their contributions prior this work.

REFERENCES

Bizer, C., Heath, T., and Berners-Lee, T. (2009). Linked data-the story so far. *Semantic Services, Interoperability and Web Applications: Emerging Concepts*, pages 205–227.

Bobadilla, J., Ortega, F., Hernando, A., and Gutirrez, A. (2013). Recommender systems survey. *Knowledge-Based Systems*, 46:109 – 132.

Bonomi, F., Milito, R., Zhu, J., and Addepalli, S. (2012). Fog Computing and Its Role in the Internet of Things. In *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, MCC '12, pages 13–16, New York, NY, USA. ACM.

Chavan RB (2014). Environmental Sustainability through Textile Recycling. *Journal of Textile Science & Engineering*, pages 1–6.

Di Noia, T., Mirizzi, R., Ostuni, V. C., Romito, D., and Zanker, M. (2012). Linked Open Data to Support Content-based Recommender Systems. In *Proceedings of the 8th International Conference on Semantic Systems*, I-SEMANTICS '12, pages 1–8, New York, NY, USA. ACM.

Figueroa, C., Vagliano, I., Rocha, O. R., and Morisio, M. (2015). A systematic literature review of Linked Data-based recommender systems. *Concurrency and Computation: Practice and Experience*, 27(17):4659–4684.

Goh, K. N., Chen, Y. Y., and Lin, E. S. (2011). Developing a smart wardrobe system. In *2011 IEEE Consumer Communications and Networking Conference (CCNC)*, pages 303–307.

Gubbi, J., Buyya, R., Marusic, S., and Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7):1645–1660.

Heitmann, B. and Hayes, C. (2010). Using Linked Data to Build Open, Collaborative Recommender Systems. In *AAAI spring symposium: linked data meets artificial intelligence*, pages 76–81.

Ingvaldsen, J. E., Ozgobek, O., and Gulla, J. A. (2015). Context-aware user-driven news recommendation. In *3rd International Workshop on News Recommendation and Analytics (INRA 2015)*. Ceur-ws Vol-1542.

Jagadish, H. V., Gehrke, J., Labrinidis, A., Papakonstantinou, Y., Patel, J. M., Ramakrishnan, R., and Shahabi, C. (2014). Big Data and Its Technical Challenges. *Commun. ACM*, 57(7):86–94.

Jannach, D., Zanker, M., Felfernig, A., and Friedrich, G. (2010). *Recommender Systems: An Introduction*. Cambridge University Press.

Klepp, I. G. and Laitala, K. (2016). Clothing consumption in Norway. Technical report 2, Oslo and Akershus University College, Oslo.

Ling, S., Indrawan, M., and Loke, S. W. (2007). RFID-Based User Profiling of Fashion Preferences: Blueprint for a Smart Wardrobe. *Int. J. Internet Protoc. Technol.*, 2(3/4):153–164.

Liu, S., Feng, J., Song, Z., Zhang, T., Lu, H., Xu, C., and Yan, S. (2012). Hi, Magic Closet, Tell Me What to Wear! In *Proceedings of the 20th ACM International Conference on Multimedia*, MM '12, pages 619–628, New York, NY, USA. ACM.

Pruit, J. C. (2015). Getting Dressed. *Popular Culture as Everyday Life*, page 155.

Schaad, P., Basler, S., Medini, M., Wissler, I., Brkle, T., and Lehmann, M. (2016). The Intelligent Wardrobe . *Studies in Health Technology and Informatics*, 225:213–217.

Tomeo, P., Fernndez-Tobas, I., Di Noia, T., and Cantador, I. (2016). Exploiting Linked Open Data in Cold-start Recommendations with Positive-only Feedback. In *Proceedings of the 4th Spanish Conference on Information Retrieval*, CERI '16, pages 11:1–11:8, New York, NY, USA. ACM.

Toney, A. P., Thomas, B. H., and Marais, W. (2006). Managing Smart Garments. In *2006 10th IEEE International Symposium on Wearable Computers*, pages 91–94.