Horizon 2020

fashion BRAIN project

Understanding Europe's Fashion Data Universe

Requirement Analysis Document WP1

Deliverable number: D1.2

Version 3.0

Funded by the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 732328

Project Acronym:	FashionBrain
Project Full Title:	Understanding Europe's Fashion Data Universe
Call:	H2020-ICT-2016-1
Topic:	ICT-14-2016-2017, Big Data PPP: Cross-sectorial and cross-lingual data integration and experimentation
Project URL:	https://fashionbrain-project.eu

Deliverable type	Report (R)
Dissemination level	**Confidential** (C)
Contractual Delivery Date	31/06/2017
Resubmission Delivery Date	27/02/2019
Number of pages	29, the last one being no. 23
Authors	Alan Akbik, Duncan Blythe - Zalando
Peer review	Benjamin Winter - BEUTH

Change Log

Version	Date	Status	Partner	Remarks
1.0	03/07/2017	Final	Zalando	Rejected 15/03/2018
2.0	30/06/2018	Resubmitted Final	Zalando	Rejected 15/10/2018
2.1	15/02/2019	Revised Draft	Zalando	
3.0	27/02/2019	Resubmitted Final	Zalando	

Deliverable Description

This deliverable will consist of a requirement analysis document as results of Task 1.2. It will collect requirements from key stakeholders from the Zalando fashion value chain.

Abstract

This document gives an overview of scenarios and research challenges in FashionBrain, based on interviews with stakeholders and data scientists at Zalando. We identified unstructured and semi-structured text data both as primary data type of interest for advanced text analytics, as well as a potential source to drive innovation in the fashion sphere. Accordingly, this document focuses on research challenges that involve text data and illustrates powerful next-generation applications that this research will enable. This overview shows important business use cases for selected external and internal customers and stakeholders. The report may serve the consortium as a source for designing exciting products and powerful next generation tools.

Important note: This deliverable is marked as **confidential** in the grant agreement, i.e. it may only be shared within the consortium, but not distributed to third parties without our explicit approval.

Table of Contents

List of Figures

List of Acronyms and Abbreviations

5W1H	What, Who, Where, When, Why, How
AI	Artificial Intelligence
CT	Core Technology
ETL	Extract, Transform, Load
GUI	Graphical User Interface
HDFS	Hadoop Distributed File System
IoT	Internet of Things
LSTM	Long Short-Term Memory
NEL	Named Entity Linking
NER	Named Entity Recognition
NNS	Nearest Neighbour Search
OEM	Original Equipment Manufacturer
OLAP	Online Analytical Processing
RDBMS	Relational Database Management System
RNN	Recurrent Neural Network
SQL	Structured Query Language
UDF	User-Defined Function
WP	Work Package

1 Introduction

Keys for the success of a retailer's business model are both a deep understanding of their assortment, as well as a deep understanding of customer needs and buying intentions. However, the necessary information to drive analytics and innovation is typically fragmented across data sources and not accessible to structured analytics. For instance, properties and features of sold articles are oftentimes available only as unstructured information, be it textual descriptions by manufacturers, user sentiment expressed in reviews, or images taken in the course of content creation. Furthermore, to derive insights about trends, competitors, pricing, or failing supply chains, a huge pool of information is potentially available on the web, such as blogs posts, retailer catalogues, and search engine results, but is unstructured and scattered across sources.

Our interviews with Zalando stakeholders and data scientists have revealed that it is of major interest to fuse all these sources of information in a structured way to enrich the product catalogue for an optimal user experience, and to research novel methods for analysing unstructured data to enable new analytics and end user applications.

In particular, we identified two broad categories of innovations based on research over unstructured data, namely innovations in search and interactivity, as well as innovations in business intelligence that are of principal interest:

1.1 Innovations in Search and Interactivity

The first category describes new and exciting ways for users to interact with a product catalogue, receive personalized recommendations and advice, do exploratory search for products, and complete buying transactions. In particular, we focus on the following innovations:

- Analyzing user interactions to derive *search intentions* from data sources such as query logs, post click behaviour, query history, and other sources. If a fashion retailer understands the mood and buying intentions of searchers correctly, it may drastically enhance conversion rates and click through rates.
- Interactive and conversational exploration by means of chatbots that are either integrated into the retailing platform, or associated with a commonly used chat application such as Skype or Telegram. With advanced AI methods, the retailer can automate communication with such customer facing applications, such as Zalon.de, effectively save costs on human labour, while simultaneously improving the customer experience by making it more personalized, interactive and novel.

Both innovations share the goal of encouraging users to interact directly with the retailing platform, and to shift traffic away from traditional web search engines such as Google to our platform. We discuss *search and interactivity*-related research challenges and application scenarios in Section 2.

1.2 Innovations in Business Intelligence

The second category includes the research and development of *large-scale text analytics* to mine insights about products, trends, competitors and customer preferences from various data sources for the purpose of business intelligence. In particular, we focus on the following innovations:

- **Brand monitoring** for internal stakeholders, which entails the identification of product mentions in unstructured data, their linking (e.g. resolution) against a product catalogue, the automated mining of opinions with regards to these products and their analysis via OLAP queries against a structured representation.
- **Time-series analysis** to better understand the developments of key metrics over time, detect and predict trends, and to identify sudden shifts in customer behaviour and brand reference.

Both innovations share the goal of mining and analysing structured information from unstructured data sources, and of creating better models for understanding customers and trends. We discuss *text analytics*-related research challenges and application scenarios in Section 3.

1.3 FashionBrain Core Technologies

The FashionBrain data integration structure covers three layers, each of which can be further subdivided into a group of Core Technologies. The relationship between Deliverables, layers and Core Technologies is as follows (with the partner leader in bold):

Data Curation & Integration Layer

CT1 Semantic Integration [**UNIFR**, MDBS, BEUTH, Fashwell]

- Shared Fashion Taxonomy [**UNIFR**] D1.3
- Attribute taxonomy and integration with deep learning [**Fashwell**, UNIFR, BEUTH] D5.1, D5.2
- Semantic Integration in Entity Linking [**UNIFR**, MDBS, BEUTH] D2.1, D2.2

CT3 Crowdsourcing interfaces and quality metrics [**USFD**, UNIFR, Zalando, Fashwell]

- D3.1, D3.2, D3.3, D3.4

Execution Layer

CT2 Infrastructures for scalable cross-domain data integration and management [**MDBS**, Beuth, UNIFR]

- D1.4, D2.3, D2.4

CT4 In-Database Named Entity Recognition and Linking methods [**BEUTH**, Zalando, MDBS]

- D4.1, D4.2, D4.3, D4.4, D6.5

Application Layer

CT5 Integration of image processing and NLP [**Zalando**, Beuth, UNIFR]

- D2.5, D6.1, D6.2, D6.3, D6.4, D6.5

CT6 Advanced image recognition [**Fashwell**, Zalando]

- D6.2, D7.6

CT7 Fashion Trends Prediction [**UNIFR**, Zalando, MDBS, USFD]

- Time Series Classification and prediction [**UNIFR**, Zalando, MDBS] D5.3, D5.4, D5.5
- Fashion Influencers Detection [**USFD**, UNIFR] D3.3

1.4 Structure of this Document

This document is structured using the two categories of innovation illustrated above: We first discuss scenarios and research challenges of category I, namely improving user experience by analyzing search intentions and exploring the product catalogue with chatbots. We then discuss scenarios and research challenges of category II, namely advanced text analytics for brand monitoring and opinion mining from various data sources. We outline each scenario and illustrate use cases and stakeholders, and specifically focus on the research challenges.

2 Improving The User Search Experience

2.1 Business Case and Stakeholder

The user search experience is of key importance to a retailer. The better and easier the search allows users to identify products they may wish to buy, the higher the chances are that users will complete a purchase. Therefore, spotting intentions of users may result in a lower bounce rate, and a higher conversion rate [8, 10]. For this reason, online retailers typically have several teams focused on enhancing the user search experience. They work on topics such as query understanding, intention detection, core search services, and search quality. In addition, there are personalization teams and teams developing and improving a recommendation engine, which are in close cooperation with search.

A particular goal is to attract customers to prefer to use a retailer's search instead of Google search. Platforms such as Zalando should strive to become the main entry-point for any fashion-related search. This can be accomplished through state-of-the-art classical search, as well as novel ways for linguistic interaction with the platform, for instance through intelligent shopping assistants (e.g. chatbots).

A key requirement to make this possible is to correctly analyze a user's search intention, whether issued as a keyword query, or as a natural language statement. That is, when a customer poses text queries against a search interface, the system needs to disambiguate the (hidden) search intention, resolve it and rank results for top-N intentions on a result page. These results should be both relevant and inspiring to the users. In the case of intelligent shopping assistants, next to relevant results we might also wish to display suggestions or other types of interaction.

To illustrate the general breadth of possible search intentions, consider Broder's taxonomy of search intentions [4], which classifies traditional queries into three broad categories:

Navigational search intention The customer has likely seen the intended product result before and wants the search engine to navigate to the desired product page. The user knows more or less exactly what she/he wants and the search results should bring up the desired products or nearly similar products. It does not matter if he or she has a buying intention. An example query of this type is:

Adidas shoes

with which a user aim to navigate all products of a certain brand.

Informational search, such as to retrieve a list of products The user wants to be informed or inspired by his/her search. Example queries for this type of intention are

hot color of dresses this season

and

what can I wear with my Adidas shoes for my next trekking hike

with which a user seeks information on trends or matching outfits. This is also referred to as *exploratory search* [13].

Transactional queries (buy) Here, the intention is not to inform yourself but to buy a product. An example query for this type of intention is

best price Adidas shoes size 46

with which a user searches for a specific product of a specific brand and size.

Our search engine or chatbot must be able to correctly interpret the full range of such queries and be able to produce results that are highly relevant to the user.

2.2 Limitations of Current Approaches

Classical product search systems are based on full-text search in which both queries and products are represented by strings [5]. The matching is thus symbolic-based and resorts to classical string matching algorithms. In practice, such a system is implemented as a pipeline, consisting of a chain of processing components, such as tokenization, lemmatization, spelling correction, acronym/synonym replacement, named-entity recognition, and query expansion. Figure 2.1 illustrates this architecture.

However, our experience has shown that such a pipeline system suffers from a number of serious limitations:

- **Limited Coverage.** Such systems cannot understand queries that are out of the vocabulary defined by the product data, due to the limitations of symbolic matching. This also affects the system's ability to scale out to a different domain. For example, in order to do internationalization, it often requires developers to investigate those language-dependent components and rewrite them.
- **Inflexible, Complex and Error-Prone Architecture.** This architecture is fragile and vulnerable to error propagation: As the output of each component is the input of the next, a defect in an upstream component can easily break down the whole system. Furthermore, dependencies between components can be complicated. For instance, a component can take in and output to multiple

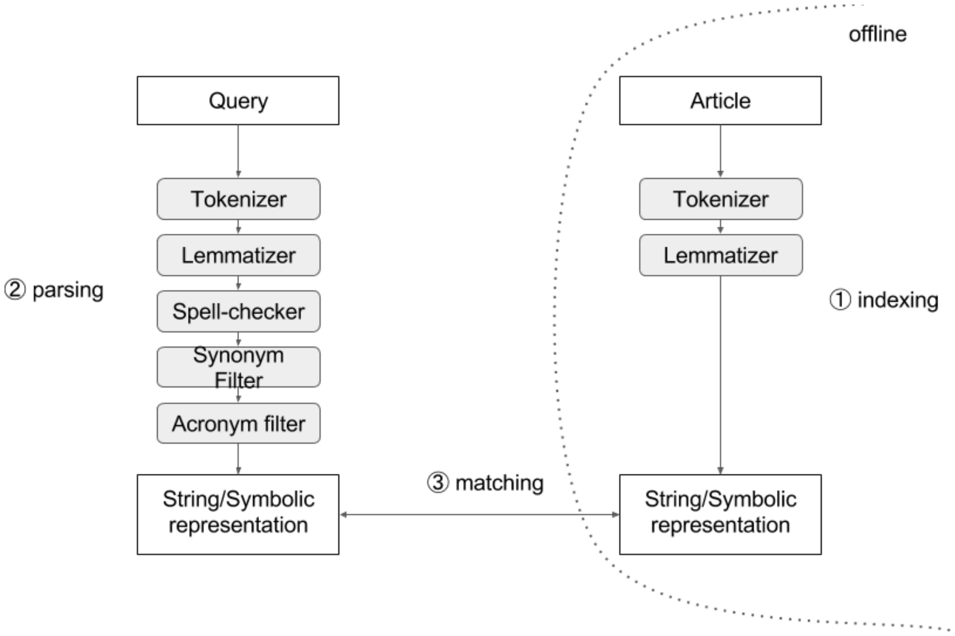


Figure 2.1: Illustration of classic full-text search architecture.

components, forming a directed acyclic graph (see Figure 2.1). In consequence, effectively maintaining and extending such a system becomes complicated. Crucially, due to the complex interactions of components, it is difficult to improve the overall search quality.

- **Limited Interactivity.** Current systems are not designed with user interaction in mind, but rather static systems that take as input a query and return a list of results. Due to their complex architecture it is unclear how to grow such systems to enable novel, innovative ways of user interaction.

This means that further research is required in order to develop approaches that address these issues. A particular requirement is the idea of an end-to-end search, which we discuss in the next section. Furthermore, we believe that more research into novel methods to enable more colloquial and interactive searches not only through a web search engine, but also through intelligent assistants will greatly improve the user search experience. In the following, we provide details for these two core scenarios.

2.3 Scenario 1: End-To-End Search

To address the limitations listed above, we require a new paradigm and philosophy to design a product search system. Specifically, our goal is to build an end-to-end product search system that directly links user search queries with products via techniques such as deep learning. This approach would eliminate all intermediate components in the pipeline, resulting in

- a simpler architecture
- more robustness and higher scalability than a pipeline approach
- a direct means for maintaining and improving the end-to-end system

One possible idea is to leverage an approach that takes as input a query, translates it into an abstract representation referred to as an *embedding*, and then matches this embedding against similar embeddings computed for products. This approach entails two challenges: The first challenge is to translate search strings into product attributes, allowing filter queries against the product catalog. The second challenge is to directly translate queries into embeddings to realize a true end-to-end search. We discuss each challenge in turn:

2.3.1 Challenge 1: Mapping Search Intentions to Product Attributes

In the first iteration, we investigate methods for analyzing the search intention of a user and determining appropriate product attributes for the user's query [11]. For instance, a user might issue the following query:

nike black sportschuhe

The system needs to analyze and interpret this query as a set of product attributes. For instance, as the following attribute-value pairs:

```
{
    "Brand": "NIKE",
    "Color": "BLACK",
    "Category": "SPORTSCHUHE",
    "Gender": "[MALE, FEMALE]",
    "AppDomain": "de"
}
```

These attribute-value pairs indicate that the above query is asking for sport shoes ("sportschuhe" in German), of black color from the brand Nike. Any gender is allowed. We can query the product catalog using these attributes and retrieve matching items.

One possible solution to address this challenge is to train a classifier to predict these attribute-value pairs. Each task corresponds to an attribute of the product and is formed as a multi-label or a multi-class classification problem where the goal is to classify a query by attribute (categorical) values. Consider the following query:

nike shoes

Our classifier must be able to predict several types of attributes from this query, namely the "brand" (which is Nike) in this query, and a "gender" (which like in the example above may be both *MALE* and *FEMALE*). We propose to train this system using query logs. That is, queries entered by users and the articles they subsequently clicked on. This provides a strong signal as to what users enter and what they expect as result. For instance, users who entered the query nike shoes typically click on results that are of the desired brand and type.

Given such data, we propose a character-based classification model which is believed to be more robust to spelling errors and variations than token-based models. A core challenge will be to gather and interpret query logs in order to train such a system, and to comparatively evaluate this approach against classical methods.

2.3.2 Challenge 2: End-To-End Learning

Strictly speaking, the system outlined in the previous point is not yet end-to-end, since it translates queries into attribute-value pairs. The limitation here is that this still requires appropriate attributes to exist in the catalog for each product. Consider the following query:

interesting shirt that I can wear to work

The user is now asking for more vague attributes, such as a shirt's style and attitude, as well as its appropriateness. Such information is problematic for standard symbolic approaches, since (1) vague information cannot be easily expressed in terms of key-value pairs, (2) it is difficult to anticipate the full set of possible queries using a pre-defined set of attributes. In fact, users may query using arbitrarily complex queries with potentially unbounded information content. This makes it necessary to go beyond pre-set attribute-value pairs in FashionBrain.

To address this challenge, we investigate neural embedding-based models to create representations of searchable objects and users intents or a user's wardrobe. Such a unified fashion item representation, like FashionDNA [3], can directly connect search intents and items in a shared embedding space and thus be a true end-to-end search.

Data sources. We investigate deriving training data from a variety of sources. This includes user query logs [9], but also potentially encompasses an analysis of user reviews (textual statements users have made with regards to fashion items), fashion blog posts and the commissioning of crowdsourcing to produce natural language data on fashion items. Ideally, these embeddings capture the full range of visual attributes (color, style, cut) as well as more abstract characteristics that can be derived from the visual representation (such as "appropriate to wear at work", "fitting for a wedding", "unique look"), meaning that the training data must reflect such information.

2.4 Scenario 2: Interactive Shopping

Google has emerged as the primary access point for every piece of information we desire. Now, text messaging is the single most popular smartphone feature, according to a Pew Research report [16]. Text-based interaction is fast, flexible, intimate, descriptive and even consistent in ways that voice and user interface often are not. The effect of all this messaging is to make us feel suddenly comfortable with what Silicon Valley has taken to calling "conversational UIs"[1] — user interfaces that you can access through text. User interfaces that scan for keywords and message you back.

Chatbots are an emerging alternative (and more human like) interface against the product catalogue (see Section 1.1). As a result, additional customers may be attracted by accessing the retailer product catalogue.

For this scenario, our principal research challenge is how to grow our end-to-end search architecture into a system that allows colloquial natural language queries and progressively enables new interactive elements. Consider for example the following complex question:

What can I wear to the opera in Hamburg in 2017 for Toska?

Such question go beyond simple keyword queries, but may be translated into an embedding similar to queries in the end-to-end search system. At the first iteration, the search engine can produce answers that directly return listings from the retailer's product catalogue. Thereby, the chatbot can leverage prior information about the style of the user, e.g. would this user prefer a sportive elegant dress or would the user rather prefer a classic elegant dress for the opera. From there, the system might be expanded to allow for clarification questions or suggestions.

2.4.1 Challenge 3: Translating Questions into Product Catalogue Queries

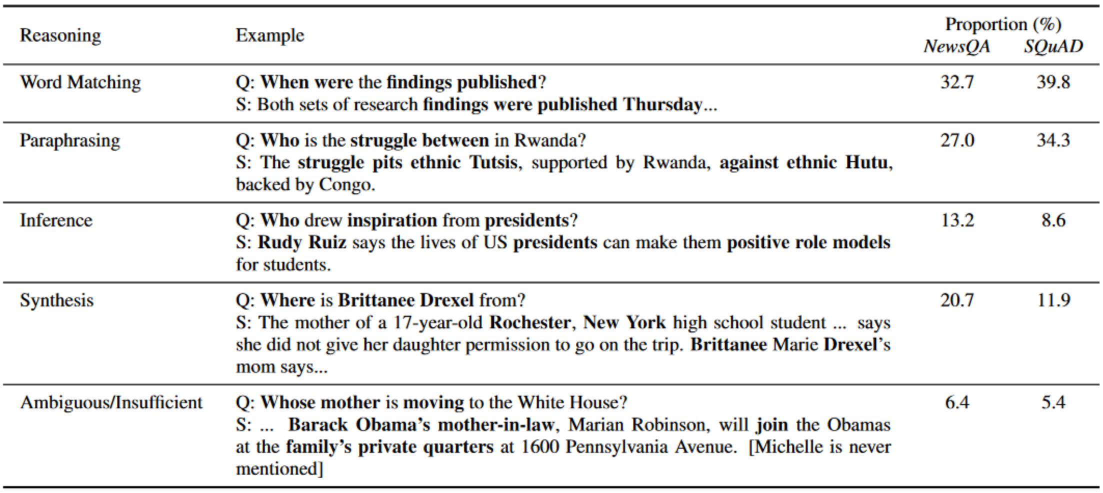
A main challenge is to translate user questions into catalogue questions. This is similar to challenge 1, but with input queries in human language, often in the form of 5W1H (What, Who, Where, When, Why, How) queries. Recent approaches try to distinguish between five types in ascending order of difficulty:

- **Word Matching**: Important words in the question exactly match words in the immediate context of an answer span, such that a keyword search algorithm can perform well on this subset.
- **Paraphrasing**: A single sentence in the article entails or paraphrases the question. Paraphrase recognition may require synonymy and world knowledge.

[1] https://venturebeat.com/2017/05/21/how-to-get-conversational-ui-right/ (accessed 6/6/17)

- **Inference**: The answer must be inferred from incomplete information in the article or by recognizing conceptual overlap. This typically draws on world knowledge.
- **Synthesis**: The answer can only be inferred by synthesizing information distributed across multiple sentences.
- **Ambiguous/Insufficient**: The question has no answer or no unique answer in the article.

Refer to Figure 2.2 for examples of each of these question types.



Reasoning	Example	Proportion (%)	
		NewsQA	*SQuAD*
Word Matching	Q: **When were** the **findings published**? S: Both sets of research **findings were published Thursday**...	32.7	39.8
Paraphrasing	Q: **Who** is the **struggle between** in Rwanda? S: The **struggle pits ethnic Tutsis**, supported by Rwanda, **against ethnic Hutu**, backed by Congo.	27.0	34.3
Inference	Q: **Who** drew **inspiration** from **presidents**? S: **Rudy Ruiz** says the lives of US **presidents** can make them **positive role models** for students.	13.2	8.6
Synthesis	Q: **Where** is **Brittanee Drexel** from? S: The mother of a 17-year-old **Rochester**, **New York** high school student ... says she did not give her daughter permission to go on the trip. **Brittanee** Marie **Drexel**'s mom says...	20.7	11.9
Ambiguous/Insufficient	Q: **Whose mother** is **moving** to the White House? S: ... **Barack Obama's mother-in-law**, Marian Robinson, will **join** the Obamas at the **family's private quarters** at 1600 Pennsylvania Avenue. [Michelle is never mentioned]	6.4	5.4

Figure 2.2: Examples of question types. (Taken from [18]).

Our challenges here are twofold: (1) Unlike for search engine queries, we do not have an existing dataset of fashion questions and results to train a Recurrent Neural Network (RNN). This data cannot be easily obtained from other sources such as user reviews or blogs, since such textual sources generally do not formulate question-answer pairs. This means that we must analyze the question-answering task in the domain of fashion and potentially produce appropriate training data. (2) Natural language queries are more complex than simple keywords. Because of this, more sophisticated models of semantics may be necessary to capture user intent and connect intent to fashion items.

A particular research and development challenge is to identify ways for progressively adding more interactivity to the existing search systems. By focusing on one type of colloquial interaction, we may be able to create prototypes that draw an initial batch of users in. These users would then start generating data through their interactions with the prototype, which we could then leverage to bootstrap more compelling interactions.

2.5 Summary of Search Challenges

In order to improve the user search experience, our main research challenges are to correctly link queries (either keyword or natural language queries) to items in the product catalogue. This becomes especially challenging for complex, informational or vague queries for two reasons: (1) Such queries become more difficult to interpret than simple keyword matching, and (2) the answer itself may be more complex than a simple retrieval of a set of attributes in the product database.

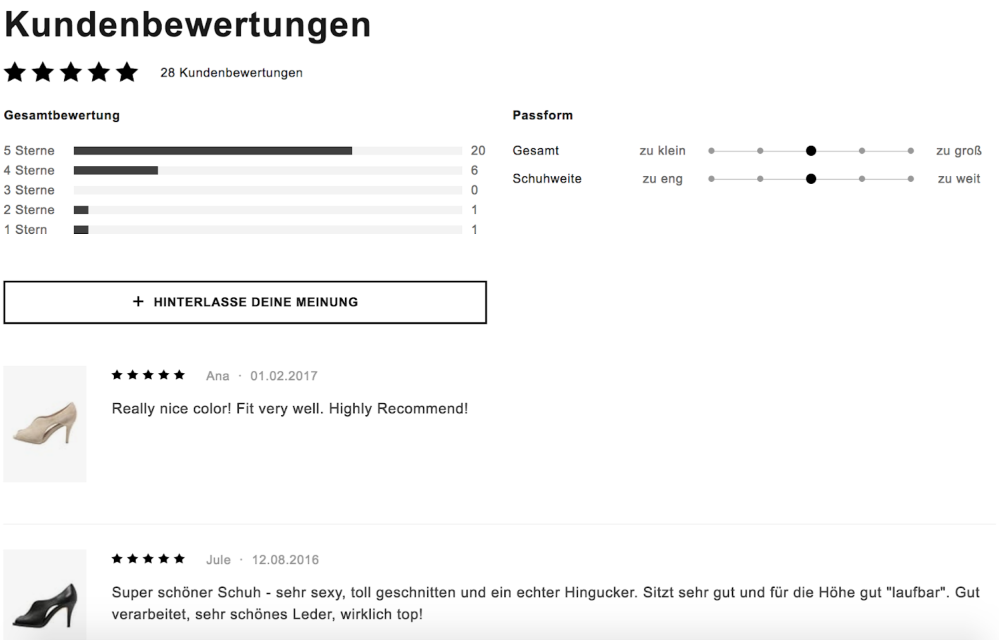
For these reasons, we propose to investigate neural architectures that map queries, natural language and products into a shared embedding space in which they can be placed in similarity relationships. Such an approach requires training data, for which we investigate both internal datasets such as query logs and user review, external datasets such as blog posts and even the creation of new datasets through methods such as crowdsourcing.

3 Text Analytics for Business Intelligence

3.1 Business Case and Stakeholder

Our interviews with stakeholders and data scientists at Zalando have pointed to a strong need to aggregate information from a range of unstructured and semi-structured data sources in order to enable business intelligence with regards to products, trends and customers. A particular challenge is that numerous data sources may contain information of interest, but often in unstructured form. Examples include blog posts in which fashion enthusiasts write about trends, brands and new products, a retailer's product catalogue that contains structured and semi-structured product descriptions, delivered by OEMs, and product reviews in which customers often write detailed reviews on specific fashion items, listing positive and negative aspects of products.

For instance, one of the most important question for creating a pleasant shopping experience in the fashion industry is working effectively with sizes and identifying products that have sizing issues. Our exploratory analysis of Zalando product reviews showed that customers tend to write about possible fitting or sizing problems



Kundenbewertungen

★★★★★ 28 Kundenbewertungen

Gesamtbewertung

5 Sterne	20
4 Sterne	6
3 Sterne	0
2 Sterne	1
1 Stern	1

Passform

Gesamt	zu klein	○ ○ ● ○ ○	zu groß
Schuhweite	zu eng	○ ○ ● ○ ○	zu weit

★★★★★ Ana · 01.02.2017

Really nice color! Fit very well. Highly Recommend!

★★★★★ Jule · 12.08.2016

Super schöner Schuh - sehr sexy, toll geschnitten und ein echter Hingucker. Sitzt sehr gut und für die Höhe gut "laufbar". Gut verarbeitet, sehr schönes Leder, wirklich top!

Figure 3.1: Product reviews left by Zalando customers.

in the product reviews. Understanding such issues and labeling products that, for instance, are of unexpected size (like "bigger than expected"), might help customers find their ideal products.

To illustrate, consider the example reviews in Figure 3.1. Here, customers make several statements about core attributes of the pair of shoes with regards to color, sizing, craftsmanship, material and even soft statements such as "ein echter Hingucker" (German for "a real eyecatcher") or "für die Höhe gut laufbar" (German for "easy to walk in despite the height"). Next to textual reviews, user can also rate the product along several dimensions, such as "overall impression" and specific size issues.

These examples illustrate how textual data may contain information that may be suitable to be added to the fashion product catalogue or for business analytics. The retailer may enhance data quality and freshness with additional data from the Web or its own review boards. Such information is useful for both internal and external stakeholders:

- **Internal stakeholders**, such as marketing or wholesales, may get a better picture of the customer needs and may proactively reorganize logistics or supply chains. For instance, analysts may want to observe a part of the Web and spot peaks in the communication behaviour at a specific date/time, and drill down to the (set of) text sources. Relevant entities are brands, locations, fashion patterns, colors, people, materials, occasions, garments, etc.
- **External stakeholders**, such as search or chatbots, may benefit from this additional representation in improving the search quality or in resolving even complex questions.

For both applications, the underlying research challenges lie in the space of entity recognition and linking, opinion mining [14], as well as OLAP queries over textual databases [12, 15].

3.2 Limitations of Current Approaches

Current approaches for business analytics still rely on manual research by fashion domain experts (on new trends and current styles), as well as traditional user interactions such as asking users to fill out questionnaires. However, as previously illustrated this approach does not easily scale to large amounts of unstructured data, since there is an inherent limit on how much text data a domain expert can consume and the breadth of feedback users are prepared to give in questionnaires. In this context, we see large-scale text analytics over various data sources as a tool to enable better decisions by experts, by helping them identify and aggregate all relevant structured information.

3.3 Scenario 3: Brand Monitoring

Specialists in Zalando are interested in analysing ‘signals’ from (in-house or external) customer reviews or blogs, specifically with regards to brands and opinions regarding these brands. Such information may be highly useful to guide stocking and advertising decisions, and to improve customer service. This scenario involves a number of research challenges.

3.3.1 Challenge 4: Linking Entities to Product Catalogue

In order to be able to aggregate references to the same product across sources, we require a robust entity linkage from text data to a structured representation in the product catalogue. At a high level this includes four sub tasks: (1) a robust entity mention detection function for recognizing entity candidates in text, (2) learning a similarity function that matches entities in the product catalogue to entity representations in text, such as blogs, (3) efficiently executing a top-N right outer join between text data and product catalogue with help of the similarity function and (4) continuously and iteratively adopting the similarity function to new product representations. For each task we will describe problems and solutions:

3.3.1.1 Subtask 1: Named Entity Recognition in Text Data

Given a text that includes characters representing fashion products, such as sentences in a blog or a review, we require a function that spots mentions of fashion products. The output of the function is a mention, denoted as a span in characters in the text. In addition, the output could also be an embedding or a form of neural representation that can be further processed in an end-to-end system for entity linkage or search.

Detecting such mentions is a well researched problem. Recent work has shown that bi-directional Long Short Term Memory (LSTM) Networks in a stacked deep learning architecture can successfully capture most mentions. We propose to build on our previous work [2, 1] in which we train character level models using a bi-directional LSTM for representing context of a single sentence of each mention.

Further research may add sentence embedding representing the context of ‘nearby’ sentences or document embeddings, representing the topic and context of an entire document. However, these approaches only focus on product mentions and ignore in-document co-references. As a result, for reaching a very high recall, we require to solve this problem.

3.3.1.2 Subtask 2: Similarity Function for Named Entity Linkage

This subtask takes as input text data and named entity mentions as well as a structured representation, such as a database table, that represents entities. The goal is to learn a similarity function for matching entities in the text data to entities from the tables. Because of the nature of the problem, the matching function needs to resolve homonyms (such as "IBM" as company based in New York or the small vendor for electronic products "Itty Bitty Maschine" company or the IATA airport code IBM), synonyms (such as "Big Blue" and "IBM") or hypernyms (such as "IBM Deutschland GmbH" vs "IBM Inc."). For a given mention in the text, the similarity function returns a list of top-N entities from the structured representation, such as a table. It depends on the user task if such a list should be limited to a Top-1 semantic or if it should be pruned at a higher order value of N.

We abstract this problem as a classification problem. Given a representation in a database table and another representation of a mention in a text. We seek to learn a function that transforms from each database record (here a product catalogue) to mentions in the text. Thereby, we can represent attribute values, schema and type information from the database into a neural embedding. Similarly, for each text where we spot a candidate mention, we abstract the context of this mention as an embedding. Representing textual context can happen at character, word, sentence, paragraph or document level. For example, the authors of [17] represent mentions at the level of a sentence. Finally, we map embeddings to structured data and text data in the same vector space.

Next, we learn a transformation function from relations to text. Learning such a function is analogous to the problem of de-duplication. Hence, we require a labeled dataset where character sequences in text data are mapped to entity representations in structured data, such as a database entry. Often, such a training set is orders of magnitudes smaller than the existing product catalogue. The job of the classifier is to generalize from the variance in the initial training set to 'unseen' representations of products and their mappings.

For reducing efforts for labeled data, we can leverage 'pre-clustering' methods to group similar representations. Such clusters can reduce labeling efforts, hence we might show a labeling user only instances of the same cluster. Another method is to apply a form of active learning in which data labeling and model training are done in parallel to reduce labeling efforts. Finally, we could also assume that learning homonyms is less error prone. Under this assumption we might generate initial labels with an exact match strategy and learn a classifier from these synthetically generated labels.

3.3.1.3 Subtask 3: Efficiently executing outer joins on top-N similarity results

In a naive fashion, the system needs to compare each mention in a text with each mention in a database. This task, which is basically a cross product, is computationally expensive. Rather, the system would take a set of mentions and would retrieve a list of likely similar entities from the table. Retrieving such a list of likely candidates is a Nearest Neighbour Search (NNS), a fundamental and essential operation in applications from many domains, such as databases, machine learning, multimedia, and computer vision. For each mention, the system must determine how many candidates it should retrieve from the NNS and to how many candidates it should prune the list after processing candidates with the similarity function from (2).

3.3.1.4 Subtask 4: Iteratively refining similarity functions

Often, new entity representations are formed in text or are represented in structured data, for which no match in the similarity function could be generated. In such cases, the system should suggest these entities for additional labeling.

3.3.2 Challenge 5: Opinion Mining on Fashion Reviews

For creating pleasant shopping experience, fashion industry retailers should work effectively with customer reviews. It is important to monitor customer feedback for specific issues (like delivery service or items quality) and also enrich product description with new qualities that customers give.

One of the most important issues to be tracked from customer reviews are sizing issues. Customers tend to write about fitting or sizing problems in the product reviews. Understanding these issues and labeling products like having unexpected size (like “bigger than expected” or “doesn’t fit well”), might help them while making an order. Product might have different dimensions and language used for describing sizing issues (“tight near the neck”, “too narrow”, “short”). Understanding this language and labeling products accordingly (“many customers” found this shoe a size bigger than usual) would lead to a better shopping experience.

It is also important to enrich product descriptions. A retailer collects and curates product descriptions, delivered by OEMs and from additional sources. However, such data may be incomplete, outdated or may not match with the desire the user has in mind. For example, the information if a pair of shoes goes well for an opera visit is subjective and is often not explicitly stated in the Original Equipment Manufacturer (OEM) description of the shoes.

For these problems we assume that a retailer will offer a corpus with fashion items from their product catalogue and customer reviews from web portal. Then the task is to first define a binary classification task of each review of either containing the desired type information, or not. One example of such a class might be all reviews that contain information with regards to sizing. This classifier will allow us to identify all relevant reviews. Building on this, we progressively define additional classes, such as service, delivery, web site, support, price, size, comfort. Reviews dataset should be labeled accordingly to these classes with crowdsourcing approach for several languages. We also indent to build finer grained classifiers that identify relevant text passages (instead of classifying the entire review). A particular challenge here is how to create training data and classifiers that work for different European languages.

3.3.3 Challenge 6: OLAP Queries over Text- and Catalogue Data

Insights in fashion related events are often desired by internal stakeholders, such as specialists working in strategic areas, marketing departments or in supply chain management/logistics. These specialists seek insights for optimizing business processes or for reacting on recently appearing events. In brand monitoring, stakeholders in Zalando desire to monitor brands, products, but also events such as product recalls, new product announcements, acquisitions, company suppliers, company competitors, man made disasters, mergers and company customers.

Entities, their spans (links to sources) and additional structured data typically reside in an RDBMS where a user will hit Online Analytical Processing (OLAP) style queries against this representation. High transactional volume is not expected, rather mass data in (e.g. hourly) batches. A brand monitoring application can typically benefit from a fast columnar main memory database with powerful analytical features, such as MonetDB [7].

One approach is to enrich MonetDB's analytical features by introducing user defined (aggregation) functions (UDFs) dedicated for executing text mining tasks with pre-trained models for entity linkage or entity recognition. The INDREX-MM system has been tested for common query patterns and can resolve for the tasks of open and closed relation extraction results in seconds on a text corpus with 800K documents and 1.6 billion linguistic and syntactic annotations. A tight integration of INDREX and MonetDB with special support for text data mining tasks will provide a powerful basis for the typical OLAP queries in a brand monitoring application.

Related Systems. SYSTEM-T from IBM Research [6] is a related system that executes Structured Query Language (SQL)-ish queries on text data and utilizes a similar approach as INDREX-MM. However, this query execution currently takes place on a shared nothing cluster (Spark + Hadoop Distributed File System (HDFS)), as a result answering times are here in minutes rather seconds. Another option is executing these tasks as “Extract, Transform, Load (ETL)” outside of the Relational Database Management System (RDBMS). Here, an external ‘model’, such as learned as in a classification, would output relations and arguments. In this case, the RDBMS would be often used only to manage, join or aggregate such data.

3.4 Scenario 4: Time Series Analysis

Time series data are series of values obtained at successive times with regular or irregular intervals between them. Many fashion data, such as customer reviews, fashion blogs, social media messages and click streams, can be regarded as time series (mostly) with irregular time intervals. Taking customer reviews as an example, the raw data can be simply modelled as one long series of <timestamp, review-text> pairs. Of course, each pair needs to be annotated with meta information, such as the identity of the customer and the reviewed product. Alternatively, one might want to split this single potentially huge review time series into multiple smaller review time series, one per product and/or one per customer.

By analysing this type of data, which we will henceforth refer to as fashion time series, fashion retailers would be able to gain valuable insights of not only trends, moods and opinions of potential customers at a given moment in time, but also the changes in trends, moods and opinions of a period of time. This is a typical scenario of time series analysis. However, unlike numerical time series which are the usual focus of the time series analysis, streaming data analysis and Internet of Things (IoT) technologies, fashion time series analysis poses new challenges (Sellam et. al, 2014; 2016a; 2016b).

Fashion time series analysis calls for a unification of several technologies that will be developed in the context of this project (pertain to task T2.3):

1. Natural language processing technologies (described in Challenges 4 and 5),
2. Analytical text data-mining (described in Challenge 6), and
3. Time Series Data Processing and Visualisation (described in Challenge 7).

3.4.1 Challenge 7: Textual Time Trails

One way to detect trends, moods and opinions is to detect popular fashion terms in the raw fashion time series data, and keep track of when/where/how each term is used over time by whom. In this way, the internal stakeholders can build fashion time trails, with each time trail contains one fashion topic.

Continuous queries can be used to automatically detect new popular terms, and state changes in the trends. GUIs similar to GoogleTrends[1] can be used to visualise the fashion time trails. See Figure 3.2 for an example illustration of such analysis. Finally, the fashion time trails are again time series, on which advanced analytical queries can be performed, such as finding correlations among time series and trends prediction.

One potential avenue for research and development is to enrich MonetDB with time series oriented features that allow for further exploration of fashion time trails. This may include adding a middleware layer to facilitate continuous data ingestion for streaming/IoT data. In addition, we are discussing:

- Adding a continuous query processing ability into the database engine, which will allow running OLAP queries over text- and catalogue-data continuously. This also pertains to FashionBrain tasks T4.1 and T4.2 respectively.

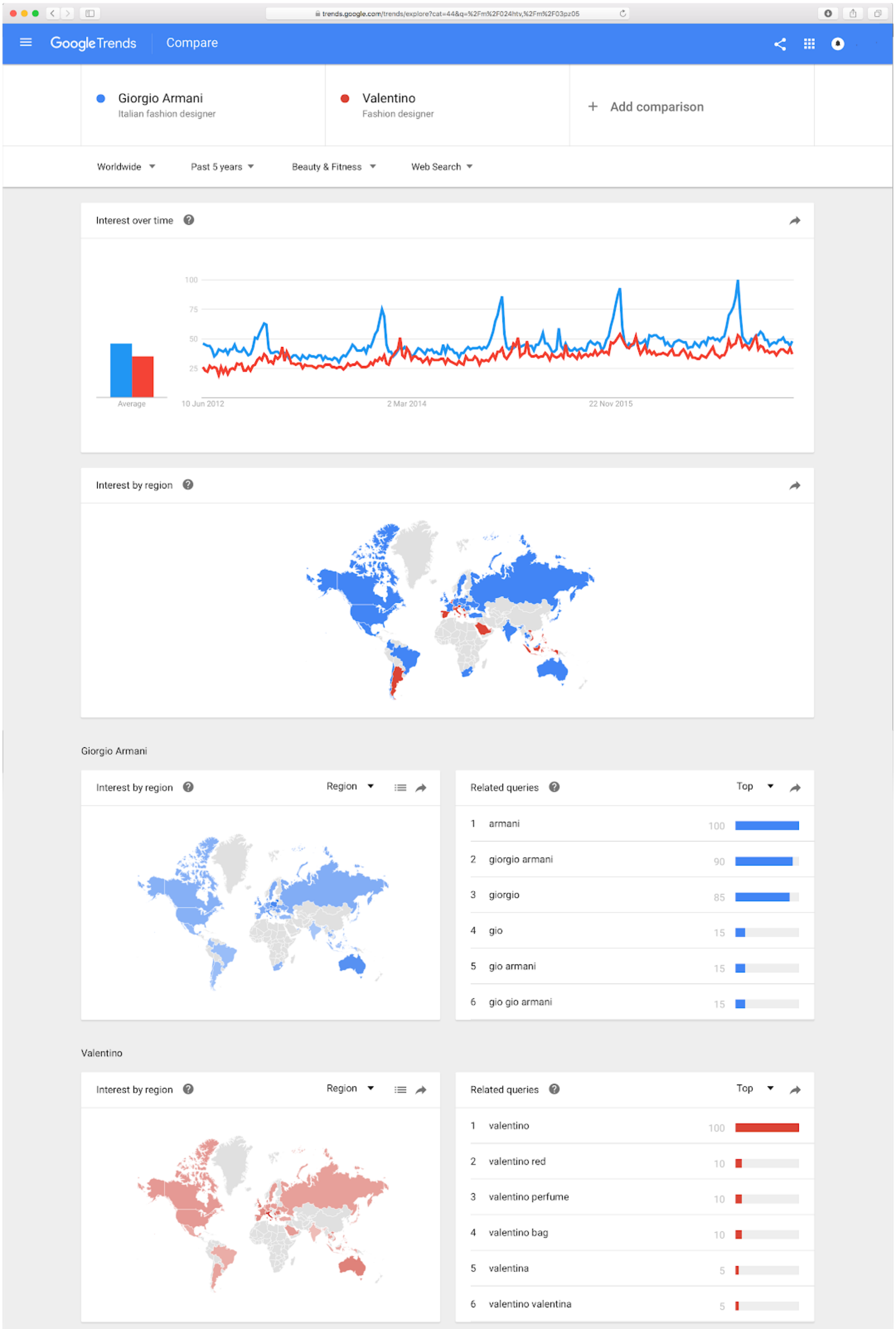
[1] https://trends.google.com/trends/

- Strengthening MonetDB's analytical power by integrating popular time series analysis libraries, such as SSM, into the database kernel. This is also relevant to FashionBrain task T1.4.
- Developing data visualisation and exploration tools for fashion time series, as pertains to FashionBrain task T2.3. For visualization, we are investigating a tight coupling between MonetDB and Grafana, the leading open source tool for time series visualization. For exploration, we will continue to develop the in-database analytics empowered data exploration tools Blaeu.

One additional challenge is how to design the UDFs in MonetDB and how to ensure a proper execution. Another challenge is learning new Named Entity Recognition (NER) and Named Entity Linking (NEL) models from data distributions which are already available in the RDBMS.

3.5 Summary of Analytics Challenges

In order to improve the business analytics with regards to customer behavior, preferences, brands and trends, we envision the research and development of powerful text analytics methods. Particular challenges include the linking of entity references in text to a database such as the product catalogue and the extraction of opinions with regards to these entities. For analytics, we distinguish between two different types: Analytics against extracted information in the form of OLAP queries against a database, and analytics that involve the processing and interpretation of time series data. We thus propose to investigate novel methods both on the level of natural language processing (entity linking, opinion mining), as well as on the systems level to facilitate large-scale analytics (OLAP and time series analytics).



Giorgio Armani – Related queries

1	armani	100
2	giorgio armani	90
3	giorgio	85
4	gio	15
5	gio armani	15
6	gio gio armani	15

Valentino – Related queries

1	valentino	100
2	valentino red	10
3	valentino perfume	10
4	valentino bag	10
5	valentina	5
6	valentino valentina	5

Figure 3.2: Example of trends in time trails.

4 Conclusion

Interviews with stakeholders and data scientists at Zalando revealed a range of research challenges with regards to unstructured and semi-structured data, as well as a number of potentially powerful next-generation applications both for customer-facing services (such as search and intelligent assistants), as well as for internal analytics. In this report, we gave an overview over such scenarios and discussed important challenges. The report may serve the consortium as a source for designing exciting products and powerful next generation tools.

Bibliography

[1] Sebastian Arnold, Robert Dziuba, and Alexander Löser. Tasty: Interactive entity linking as-you-type. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, pages 111–115, 2016.

[2] Sebastian Arnold, Felix A Gers, Torsten Kilias, and Alexander Löser. Robust named entity recognition in idiosyncratic domains. *arXiv preprint arXiv:1608.06757*, 2016.

[3] Christian Bracher, Sebastian Heinz, and Roland Vollgraf. Fashion dna: merging content and sales data for recommendation and article mapping. *arXiv preprint arXiv:1609.02489*, 2016.

[4] Andrei Broder. A taxonomy of web search. In *ACM Sigir forum*, volume 36, pages 3–10. ACM, 2002.

[5] Surajit Chaudhuri, Yi Chen, and Jeffrey Xu Yu. Guest editors introduction: Special section on keyword search on structured data. *IEEE Transactions on Knowledge & Data Engineering*, (12):1761–1762, 2011.

[6] Laura Chiticariu, Rajasekar Krishnamurthy, Yunyao Li, Sriram Raghavan, Frederick R Reiss, and Shivakumar Vaithyanathan. Systemt: an algebraic approach to declarative information extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 128–137. Association for Computational Linguistics, 2010.

[7] Arjen P de Vries, Nikos Mamoulis, Niels Nes, and Martin Kersten. Efficient k-nn search on vertically decomposed data. In *Proceedings of the 2002 ACM SIGMOD international conference on Management of data*, pages 322–333. ACM, 2002.

[8] Qi Guo and Eugene Agichtein. Ready to buy or just browsing?: detecting web searcher goals from interaction data. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 130–137. ACM, 2010.

[9] Qi Guo and Eugene Agichtein. Beyond dwell time: estimating document relevance from cursor movements and other post-click searcher behavior. In *Proceedings of the 21st international conference on World Wide Web*, pages 569–578. ACM, 2012.

[10] Qi Guo, Dmitry Lagun, and Eugene Agichtein. Predicting web search success with fine-grained interaction data. In *Proceedings of the 21st ACM international*

conference on Information and knowledge management, pages 2050–2054. ACM, 2012.

[11] Alpa Jain and Marco Pennacchiotti. Open entity extraction from web search query logs. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 510–518. Association for Computational Linguistics, 2010.

[12] Torsten Kilias, Alexander Löser, and Periklis Andritsos. Indrex: In-database relation extraction. *Information Systems*, 53:124–144, 2015.

[13] Gary Marchionini. Exploratory search: from finding to understanding. *Communications of the ACM*, 49(4):41–46, 2006.

[14] Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, AL-Smadi Mohammad, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 19–30, 2016.

[15] Rudolf Schneider, Cordula Guder, Torsten Kilias, Alexander Löser, Jens Graupmann, and Oleksandr Kozachuk. Interactive relation extraction in main memory database systems. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, pages 103–106, 2016.

[16] AUS Smith. smartphone use in 2015. pew research center. 2015, 2015.

[17] Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury, and Michael Gamon. Representing text for joint embedding of text and knowledge bases. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509, 2015.

[18] Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. Newsqa: A machine comprehension dataset. *arXiv preprint arXiv:1611.09830*, 2016.