Horizon 2020

Understanding Europe's Fashion Data Universe

Early Demo on Textual Image Search

Deliverable number: D6.3

Version 2.0

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	Demonstrator (D)
Dissemination level	**Public** (P)
Contractual Delivery Date	31/06/2018
Resubmission Delivery Date	27/02/2019
Number of pages	18, the last one being no. 12
Authors	Alan Akbik, Duncan Blythe - Zalando
Peer review	Mourad Khayati - UNIFR

Change Log

Version	Date	Status	Partner	Remarks
1.0	31/06/2018	Final	Zalando	Rejected 15/10/2018
1.1	15/02/2019	Revised Draft	Zalando	
2.0	27/02/2019	Resubmitted Final	Zalando	

Deliverable Description

This deliverable consists of a preliminary image search prototype based on textual entities. This is the basis for D6.5, which will extend the textual component by NLP and multilinguality.

Abstract

Deliverable D6.3 is a demonstrator that showcases the text-to-image search capabilities that are developed over the course of work package 6. The overall goal is to match textual search queries such as "rotes Kleid" (engl. "red dress") to a ranked list of fashion products. As key innovation, we present a neural information retrieval approach that learns to embed both textual queries and product images as real-valued vectors in a shared, high-dimensonal vector space. We train the approach so that text and images with similar semantics are embedded close to each other in this shared space, thus allowing us to compute matching text-image pairs using the cosine similarity of their respective embeddings. This report accompanies the demonstrator, which is the basis for D6.5 that will extend the textual component by NLP and multilinguality.

A video demonstrator is available online at https://fashionbrain-project.eu/early-demo-on-textual-image-search/.

Table of Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

DSSM	Deep Structured Semantic Models
GRU	Gated Recurrent Unit
NLP	Natural Language Processing

1 Introduction

As discussed in detail in the strategy deliverable D1.2, one of the core innovations of FashionBrain is to develop new information retrieval technologies so that customers can better search and find fashion items in an online shop. For instance, customers might issue a full text query such as "Schuhe von Adidas" (engl. "shoes by Adidas") and expect to find a ranked list of matching items in the product catalogue. Importantly, full text queries impose no restrictions on a customer, allowing the customer to issue any query he/she may find appropriate, including vague, exploratory and subjective queries such as "what can I wear with my Adidas shoes for my next trekking hike?". Developing technologies to handle arbitrarily complex search queries will enable us to connect more customers to products and thus increase revenue of fashion e-commerce companies.

Limitations of classic approaches. Classic approaches for information retrieval, such as the one developed by Zalando pre-FashionBrain, use a cascaded architecture to perform full-text search of the product database. The products are indexed by attributes which accompany them, either added in the course of the manufacturing process or enriched in the product curation process. The search architecture follows a cascaded design with input strings preprocessed as displayed in Figure 1.1.

In the process of this cascade, several key Natural Language Processing (NLP) tasks are implemented which lead to a structured query which can be executed on the product database. Although this approach is well-tested and understood, it suffers several disadvantages. These include but are not limited to:

- Unclear relationship between component parts and end goal of improving retrieval
- Subtasks are not strictly necessary for retrieval
- Does not easily scale to new languages without language experts and additional development efforts
- Architecture fragile and may be broken by bugs in any of component parts
- Sequence of operations is arbitrary with many recurrent dependencies
- Retrieval is restricted to attributes explicitly represented in the product database
- No image-level information encoded

For a full discussion of the limitations of classic information retrieval systems we refer the reader to deliverable D1.2.

Proposed approach. As a first step towards solving these issues this deliverable describes an alternative system which optimizes the mapping between query string

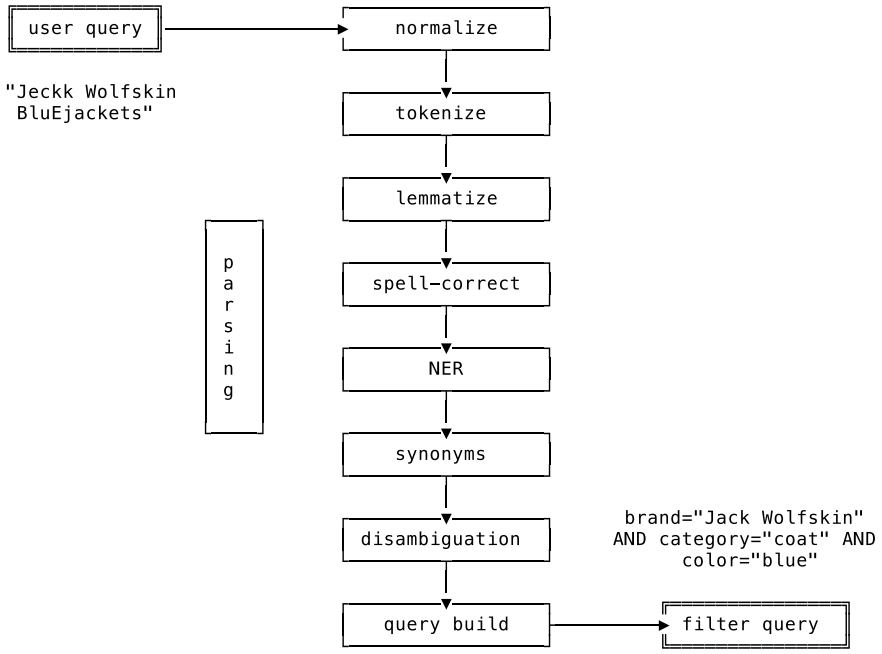

Figure 1.1: Illustration of classic, cascaded full-text search architecture.

and a product representation based on recent advances in deep learning. Rather than applying a cascade of operations we propose embedding the query string and products directly into a shared vector space in which matching products and queries have similar vectors. A red dress for instance would be embedded to a vector that is similar to a vector of an image showing a red dress and dissimilar to a vector of an image showing a black shoe. The challenge, thus, is to create a system that given text and images can compute meaningul embeddings in such a way for arbitrary textual strings and product images. We refer to this approach as *neural information retrieval*.

1.1 Placement of this Deliverable within FashionBrain

In this deliverable, *D6.3 "Early Demo on textual image search"*, we present the first prototype of our neureal information retrieal research, which we will build upon for D6.5. As such, the requirements for this demonstrator are derived from the business scenarios identified in *D1.2 "Requirement analysis document"*, in particular:

- Scenario 1: End-To-End Multi-language Search
  - Challenge 1: Mapping Search Intentions to Product Attributes
  - Challenge 2: End-To-End Learning

2 Architecture

As discussed in the introduction, we seek to embed images and text into a shared vector space that model multimodal text-image semantic similarity. To achieve this, we adopt an architecture loosely based on Deep Structured Semantic Models (DSSM) [3] and trained with a rank-loss hinge objective. This architecture defines two separate neural networks, one for each data type: The first, a recurrent neural network, is used to process text, while the second, a convolutional neural network, is used to process images. Both networks are trained to produce embedding vectors that are similar according to the cosine distance if a text matches an image. An overview of this architecture is shown in Figure 2.1.

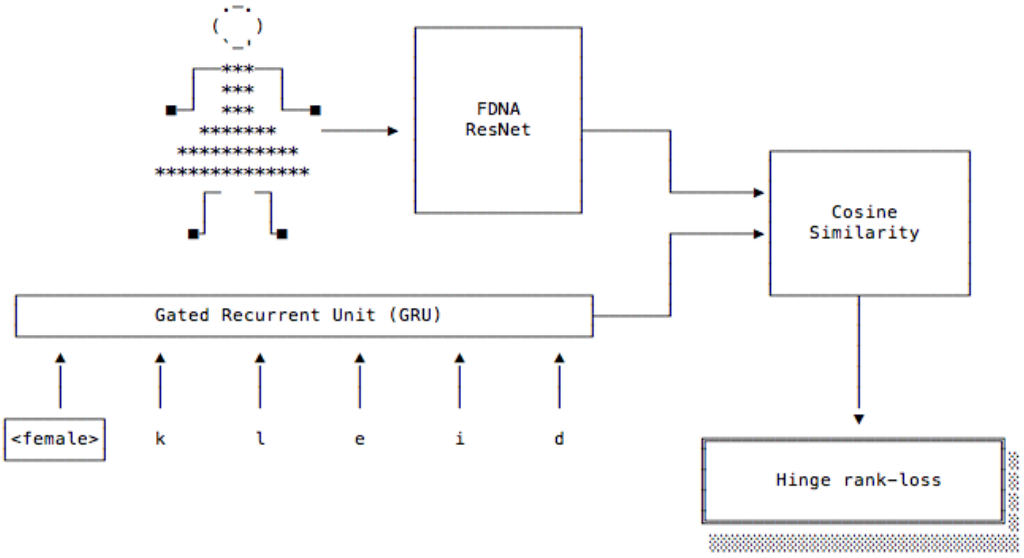

Figure 2.1: Illustration of proposed neural information retrieval approach.

Embedding text. We represent the query string (exemplified in Figure 2.1 by "kleid", which is German for "dress") as a sequence of one-hot vectors, with dimension equal to the number of characters in the alphabet we are interested in (in this case the alphabet over queries in the Germany app-domain). We add the gender category in which the user is searching to the beginning of this string. The string is then passed one-character at a time to a type of recurrent neural network, namely a Gated Recurrent Unit (GRU) neural network [1] and the final hidden state is extracted.

More precisely, our sequence of one hot vectors is

$$x_0, x_1, \ldots, x_T$$

Each one hot vector is then embedded in a vector space to give

$$u_t$$

and updated linearly using the non-linear recurrence of the GRU (f).

$$u_t = W_x x_t \; h_t = f(u_t, h_{t-1})$$

This final hidden state then represents the user-textual input.

Embedding images. On the image side, we encode all images in the current product data using a deep-residual convolution neural network (resnet) [2]. So for a product image I, the network maps to a vector $g(I)$. Given h_t and $g(I)$ we may compare the vectors and formulate an objective function which forces images and text which match to be "close" and otherwise "far" from one-another. To measure closeness we use cosine-similarity:

$$\cos(x, y) = \frac{x^\top y}{\|x\|, \|y\|}$$

Our rank objective [4] enforces matching by encouraging matching image and text to be similar to another up to a threshold λ. Let I^+ be a match to h_T and I^- a mismatching image. Then:

$$\mathcal{L}(h_T, I^+, I^-) = \max(0, \lambda + \cos(h_t, g(I^+)) - \cos(h_t, g(I^-)))$$

2.1 Model Training

We train and evaluate the prototype using past click data collected from the Zalando web site: We consider each time a customer entered a text query and then clicked on one of the search results as a positive training example. That is, if a user entered the query "rotes Kleid" ("red dress") and then clicked on a fashion item represented by an image in the search result, we assume that this fashion item image has similar semantics as the textual query. To add negative training examples, we randomly sample the dataset.

We then train our neural architecture using stochastic gradient descent (SGD) with the hinge loss function as detailed above. We use the standard training method of mini-batching and gradually reduce the learning rate during training with an annealing scheme against the training loss. We continue training until validation loss no longer improves and the learning rate has fully annealed. Refer to Figure 2.2

for a plot of the development of the loss on the training and validation folds.

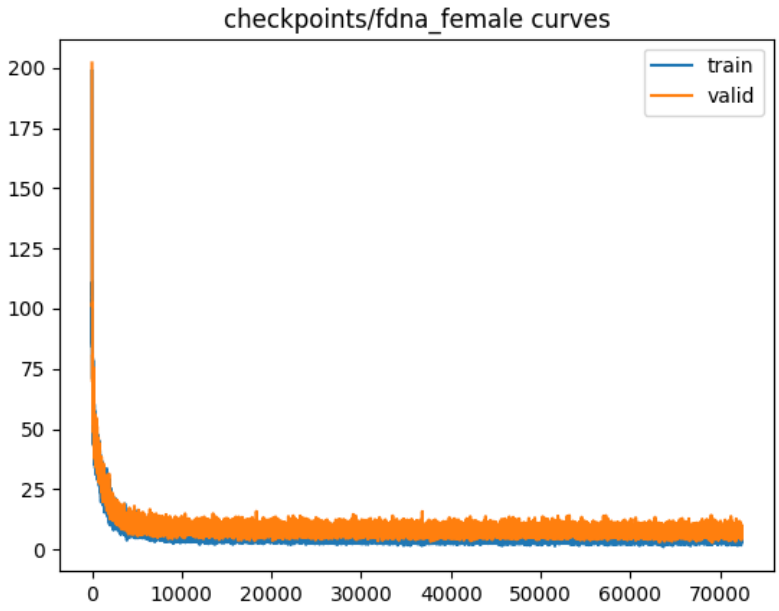

Figure 2.2: Plot of loss on the training and validation folds over training iterations.

2.2 Quantitative Evaluation

We quantitatively evaluate our approach using a held-out test set of query-image pairs and measuring the top-N precision which counts the number of correct among the top-N returned results. Refer to Table 2.1 for an overview of results for top 1, top 10, top 100 and top 1000 for different ablations of our proposed approach.

Parameters. We vary a number of hyperparameters to evaluate their impact on overall performance; In our experimentation we vary the hyper-parameter λ, the minimum number of examples required for a query to appear in training ("N") and whether we additionally include an embedding of attributes with the deep-residual features ("use_attr"). As Table 2.1 shows, we find that the model using 5 counts, only residual featues ("fdna=True") and a hyper-parameter $\lambda = 0.2$ to perform best in our prototype.

n	use attr	λ	top 1	top 10	top 100	top 1000	percentile
1	False	0.2	9.1 ± 28.76	**41.4** ± 49.25	79.3 ± 40.52	95.5 ± 20.73	1.51 ± 6.07
5	False	0.2	**10.2** ± 30.26	41.3 ± 49.24	**79.8** ± 40.15	**95.6** ± 20.51	1.47 ± 5.70
20	False	0.2	7.9 ± 26.97	36.1 ± 48.03	74.1 ± 43.81	93.8 ± 24.12	2.12 ± 7.56
5	True	0.2	8.7 ± 28.18	34.8 ± 47.63	70.1 ± 45.78	93.3 ± 25.00	1.87 ± 5.36
5	False	0.1	4.5 ± 20.73	25.6 ± 43.64	62.3 ± 48.46	91.1 ± 28.47	2.82 ± 7.59
5	False	0.3	7.7 ± 26.66	35.3 ± 47.79	76.2 ± 42.59	94.6 ± 22.60	1.79 ± 6.45

Table 2.1: Overview of evaluation results.

2.3 Qualitative Evaluation

This deliverable is a demonstrator of the proposed approach in which we can type a textual query in German and retrieve a ranked list of product images. We use it to qualitatively explore whether our approach yields good query results. When evaluating our approach, we note that a good fashion product search system should fulfil the following desiderata:

1. Retrieving items based on basic attributes such as color, silhouette, brand
2. Meaningfully use gender as a basis for restricting the search domain
3. Be robust to spelling errors
4. Be able to work with abstract concepts over and above simple attribute types
5. Suggest meaningful products even when a requested item is not present in the database

With these desiderata in mind, we define some example queries which we discuss here.

Basic queries. We issue some basic queries that use combinations of product type, color and color-concept. For instance, the query "Kleid" (engl. "dress") yields results as shown in Figure 2.3, results for "rotes Kleid" (engl. "red dress") are shown in Figure 2.4 and results for "buntes Kleid" (engl. "colorful dress") are shown in Figure 2.5. As the figures show, our approach is well suited to identifying colors and even color concepts such as "colorful".

We also issue the query "Nike shoes", which is one of the example queries we discussed in the strategy deliverable D1.2. As Figure 2.6 shows, the approach is well suited to identifying concepts such as brands and item types.

Complex queries. We also test the prototype for more complex search queries. As discussed above, we aim for an approach that is robust to spelling errors. As shown in Figure 2.7, even a heavily misspelled query for "Adidas" still yields fitting results due to the character-level model we use to encode text queries.

We also issue the abstract query "Büro", which means "office" in English. As Figure 2.8 shows, search results now list attire suited to be worn in an office environment. Similar, when issuing the query "Fußballhemd", which means "shirt

for soccer" in English, we get soccer related results, many of which are shirts to be worn for soccer. These examples illustrate that our approach is suited to going beyond simple, descriptive queries and can potentially capture more complex and abstract semantics.

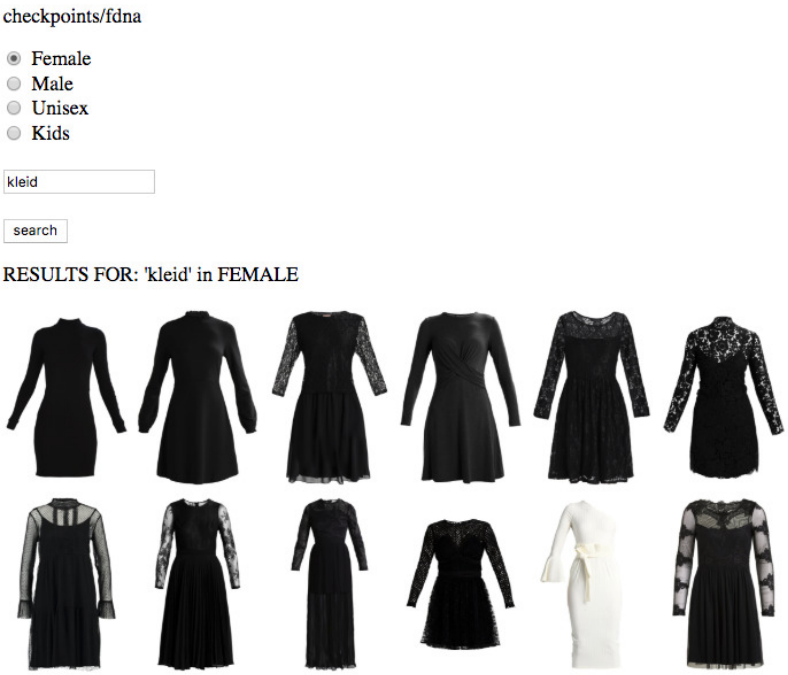

Figure 2.3: Search results for "Kleid", which is "dress" in English.

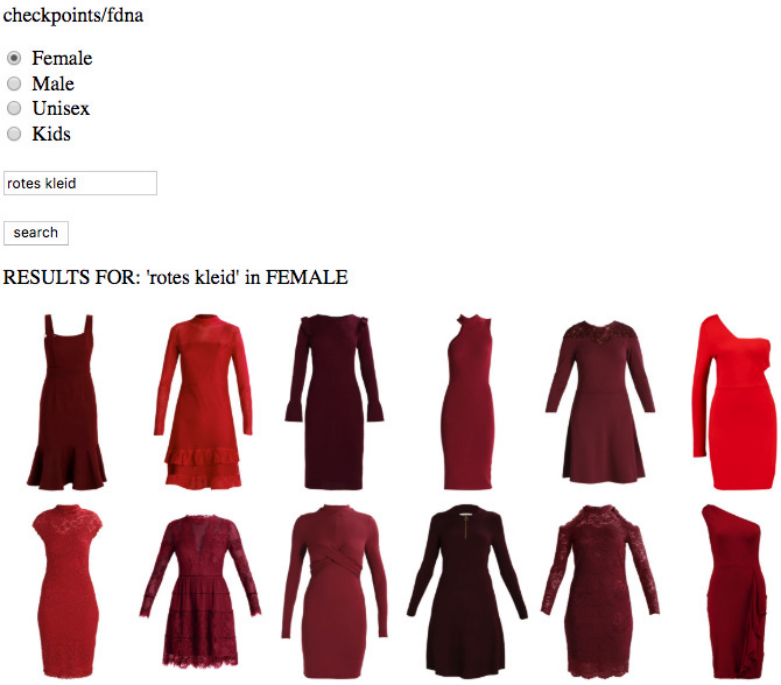

Figure 2.4: Search results for "rotes Kleid", which is "red dress" in English.

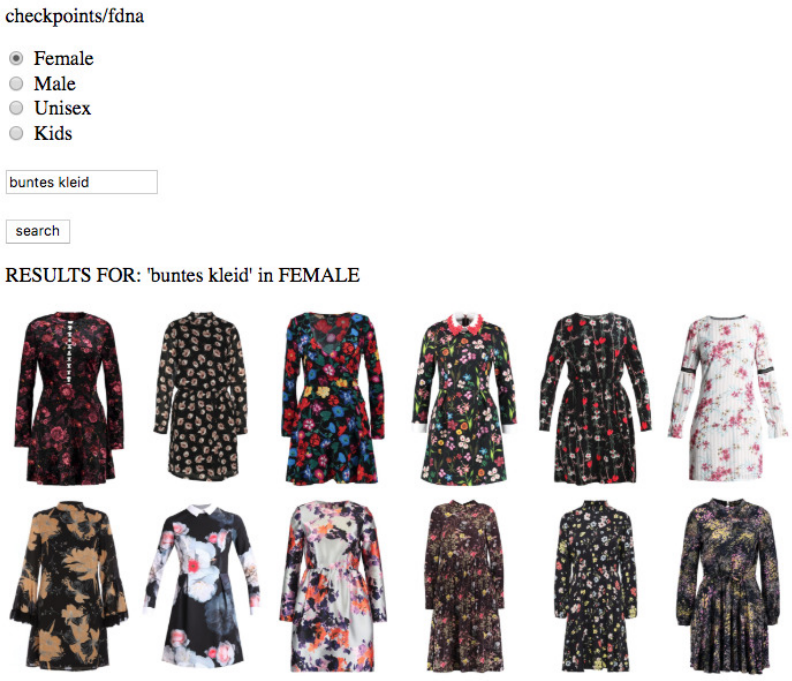

Figure 2.5: Search results for "buntes Kleid", which is "colorful dress" in English.

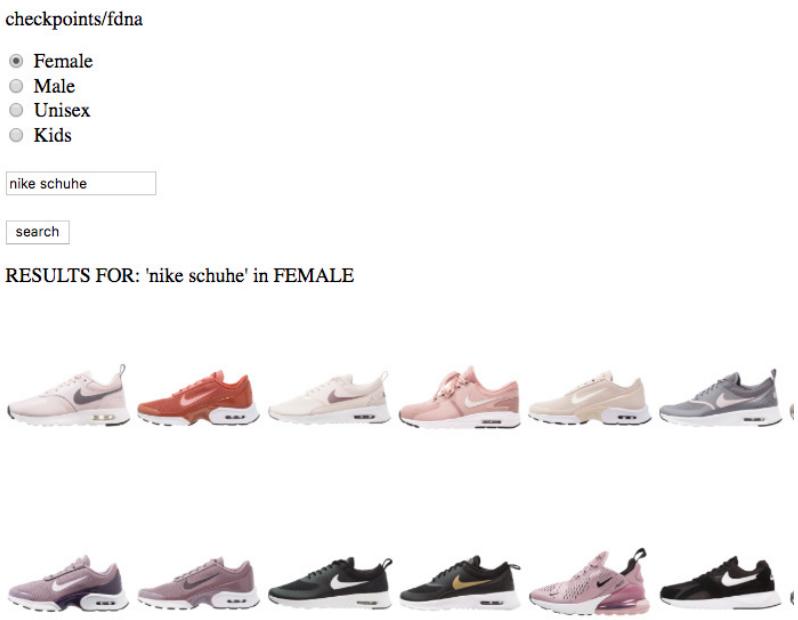

Figure 2.6: Search results for "Nike Schuhe", which is "Nike shoes" in English.

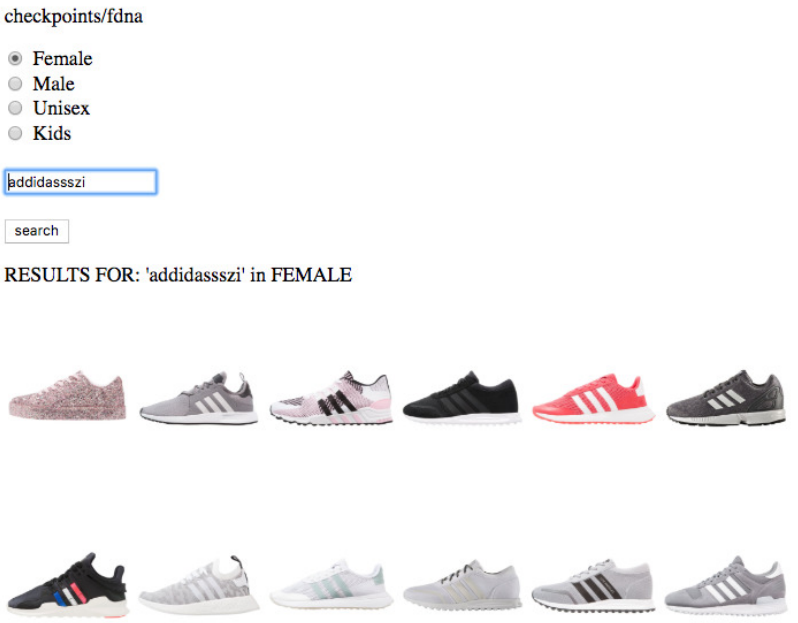

Figure 2.7: Search results for heavily misspelled "Adidas" query. The search engine can nevertheless return appropriate results.

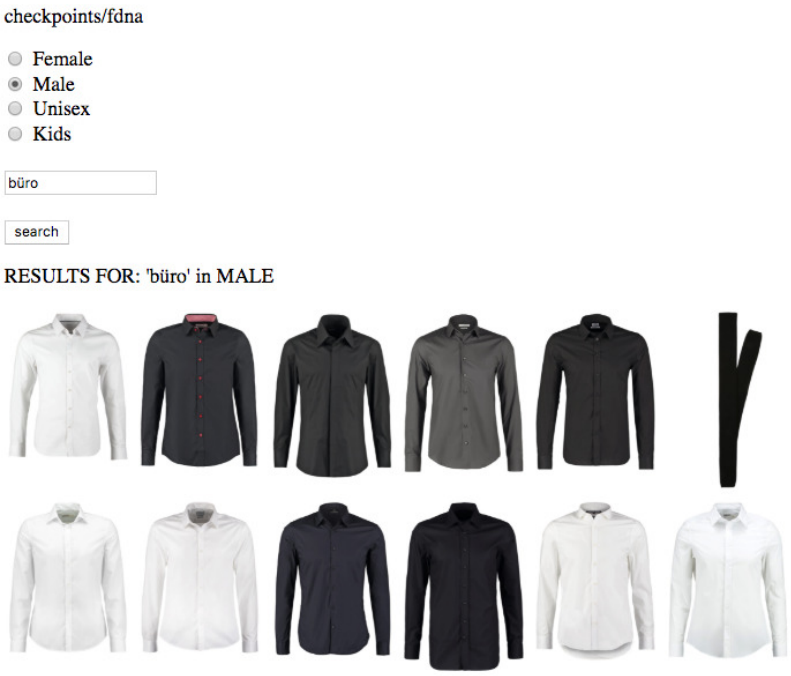

Figure 2.8: Search results for "Büro", which is "office" in English.

Figure 2.9: Search results for "Fußballhemd", which is "soccer shirt" in English.

3 Conclusion

We presented the first prototype of our neural information retrieval system designed to match textual queries to images. Our prototype is operational for the German language and is currently being extended to increase the breath of semantics it captures. We make the prototype available as a video demonstrator of the FashionBrain web site, available at https://fashionbrain-project.eu/early-demo-on-textual-image-search/. Deliverable D6.5 will present the final, extended iteration of this system.

Bibliography

[1] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[3] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 2333–2338. ACM, 2013.

[4] Andrew Trotman. Learning to rank. *Information Retrieval*, 8(3):359–381, 2005.