

Understanding Complex Casual Leisure Information Needs: An Analysis of Search Requests for Books, Games, Movies and Music

Toine Bogers ^{1,4*}, Maria Gäde ², Marijn Koolen ³, Vivien Petras ² and Mette Skov ⁴

¹ IT University of Copenhagen, Copenhagen, Denmark

² Humboldt-Universität zu Berlin, Berlin, Germany

³ KNAW Humanities Cluster, Amsterdam, the Netherlands

⁴ Aalborg University, Aalborg, Denmark

* Author to whom correspondence should be addressed; tobo@itu.dk. All authors contributed to the paper equally.

Abstract:

Purpose. In this paper we introduce the CRISPS (CROSS-domain relevance aSPects Scheme) coding scheme for complex information needs in the four leisure domains of Books, Games, Movies and Music. Our findings can be used to prioritize efforts to help existing search engines better support complex information needs, both by prioritizing which aspects are easier to classify automatically and by determining which information sources should be considered.

Design/methodology/approach. A cross-domain classification scheme for relevance aspects and information needs in casual leisure domains (CRISPS) is developed and applied. The paper provides the documentation of the scheme development and annotation process as well as a detailed, large-scale analysis of 2000 requests (500 per domain) and relevance aspects for four domains, as expressed in complex search requests in everyday-life information seeking posted to online forums.

Findings. We identify and discuss relevance aspect frequencies, information need types, and the described search process of the requests. Furthermore, the coding scheme development and annotation process is documented and reflected on.

Originality/value. This is the first categorization and analysis of complex information needs in these four leisure domains combined. The coding scheme and findings can be used to develop new types of search interfaces that incorporate the kinds of relevance aspects identified in the scheme allowing to express complex needs in the form of structured queries.

Keywords: complex search tasks, cross domain, information need, relevance aspects

A. Developing the Coding Scheme

This appendix provides more details on the development of the coding scheme for the four leisure domains with information on data gathering, processing and the axial coding process.

The scheme development happened in consecutive stages as domains were added for comparison. First books and movies were analyzed, followed by games and finally music. Data from three different sources was either collected for this project or for previous projects and re-used¹.

A.1. Data collection

1.1.1. Books

Book requests were collected from the online discussion forums of LibraryThing (LT), one of the primary social cataloging websites for books². Our data collection crawled LT forum threads in 2012, which resulted in 115,858 XML-converted posts. Those were automatically filtered to include typical search request expressions, such as ‘*suggest*’, ‘*looking for*’ and ‘*which books*’, which resulted in a sub-sample of 1,461 requests³. For scheme development, we considered the first post in these threads, which contained the original request.

1.1.2. Movies

For movie requests, the now defunct Internet Movie DataBase (IMDB) message boards were crawled. The IMDB message boards were crawled in June 2014 and February 2017, just before their shutdown. While the exact number of available message boards and threads could not be determined at the time of the crawl, in 2015, they contained over 1.19 million threads³. From the IMDB message boards, we crawled two, which were most likely to contain search requests for movies: ‘*I need to know*’ (INTK), where most threads were concerned with re-finding a specific movie and ‘*Lists & recommendations*’ (LaR), which contained threads interested in movie recommendations or lists of similar movies with a particular theme. For the analysis here, the non-overlapping crawls were combined to 6,320 INTK threads and 634 LaR threads, resulting—after data pre-processing—in 6,879 XML initial posts, potentially containing movie requests. Because the posts were not filtered like the book threads, not all of them contained actual movie requests. The 2014 crawl was analyzed previously³ and contained 68.8% movie search requests. The type of post was determined during coding and only requests were considered.

1.1.3. Games

For game search requests, three discussion groups for video game-related information needs on Reddit³ were selected³. The subreddits ‘*gamingsuggestions*’ and ‘*gamesuggestions*’ contained requests for new

¹ The data is available at <https://doi.org/10.5281/zenodo.6814288>.

² <http://www.librarything.com/>, last visited February 4, 2022.

³ Available at <http://reddit.com>, last visited February 5, 2022.

games to play, while *'tipofmyjoystick'* contains known-item (i.e. re-finding) requests for video games. A Reddit crawler⁴ was used to crawl all posts to the three subreddits from June 2-22, 2018, a total of 2,266 threads. About half of the threads were in the *'tipofmyjoystick'* known-item subreddit, the other half in the other two subreddits on game suggestions. The *'tipofmyjoystick'* subreddit for known-item requests suggests a template for formulating an information need, examples in this subreddit appear more structured and standardized than in other forums and may have influenced the scheme development.

1.1.4. Music

Music requests were extracted from Reddit with the same crawler and within the same time frame as the games requests. We extracted all threads from June 2-22, 2018 from three subreddits: *'namethatsong'* and *'whatsthat song'* contained known-item requests, whereas *'musicsuggestions'* contained requests for music recommendation. After data preprocessing, a total of 1,044 threads was considered. The music dataset heavily skews towards know-item requests: only 91 posts came from the *'musicsuggestions'* thread.

A.2. Axial Coding

1.2.1. Books and movies.

The coding schemes for books and movies were developed simultaneously, but separately for each domain. Each of the five authors derived their own codes for both domains from a random set of book and movie requests through an inductive coding approach in order to maximize different perspectives on the categories ?. The requests included the title, the full text of the first post and the group it was posted in. The size of the development sets of requests that each author went through (50 for books, 75 for movies) was selected based on the estimated frequencies of even rare relevance aspects occurring in the number of requests and was based on earlier coding experiments ?. The first round of development resulted in 89 suggested relevance aspects for books and 82 for movies, organized in different category schemes by each author.

In the second round, we used card sorting (digitally and in a face-to-face setting) in different author combinations to merge the five different category schemes for each domain into one scheme for the domain, which contained fewer, mostly non-overlapping categories.

The third round comprised grouping related categories into top-level aspects and agreeing on the labels for the categories. This was also done in several iterations by different author combinations until all agreed on the final scheme for each domain. It is important to note that we finalized the book category scheme first, which influenced the categories and labels of the movie scheme.

In the final and fourth round, similar aspects in both domains were identified and grouped and labeled the same, so that the schemes could be joined and compared with each other. This also impacted the book category scheme when an aspect in the movie scheme encompassed one in the book scheme, for instance

⁴ Available at <https://github.com/lucas-tulio/simple-reddit-crawler>, last visited February 6, 2022.

‘Contributor’ encompassing ‘Author’. For each aspect and information need, we added a scope note to describe their content as well as examples from the development sets of forum requests.

1.2.2. Games and music.

For developing the relevance aspect schemes for games and music, we also developed independent category schemes for each domain, although the categories and labels were definitely influenced by the previously analyzed domains (books and movies). Music and games generated a number of different categories, we had not encountered in the other two domains, however. The deductive coding approach helped in discovering these different categories. For each domain, we selected random samples of 75 threads from the Reddit crawls. For music, the initial round of development resulted in three different coding schemes with a combined total of 90 codes. For games, it was 95 codes, the highest total number, which also represents the apparent complexity in this domain.

Two authors used card sorting to merge the individual category schemes into a unified scheme for each domain, which was then discussed with the whole group until consensus was reached. The following iterations closely followed the process for the other domains: related categories were grouped into higher-level categories and scope notes and examples added.

B. Inter-Annotator Agreement

This appendix describes Inter-Annotator-Agreement (IAA) in more detail, per relevance sub-aspect. The per sub-aspects IAA scores are presented in multiple tables, one per main category.

Table 1. Inter-Annotator Agreement for the Content sub-categories across all four domains.

Content	Books		Games		Movies		Music	
	IAA	N pos	IAA	N pos	IAA	N pos	IAA	N pos
Character(S)	0.55	32	0.72	27	0.44	40	N/A	
Cutscene(S)	N/A		0.38	4	N/A		N/A	
Dialogue & Lyrics	-0.01	2	0.85	4	0.88	5	0.85	27
Gameplay Mechanics	N/A		0.69	55	N/A		N/A	
Graphic Design	0.90	13	0.76	28	-0.03	3	N/A	
Instrument(S)	N/A		N/A		N/A		0.58	11
Melody	N/A		N/A		N/A		0.85	4
Plot	0.86	35	0.35	27	0.90	56	0.25	6
Rhythm & Tempo	N/A		N/A		N/A		0.70	18
Setting	0.49	28	0.09	22	0.72	17	N/A	
Sound Design	N/A		0.65	4	N/A		0.19	13
Structure	0.11	11	-0.03	5	-0.01	1	0.58	11
Time	0.62	16	-0.01	1	0.20	7	N/A	
Topic	0.69	49	-0.05	8	0.42	12	0.48	11
Vocals	N/A		N/A		N/A		0.63	22
World Building	-0.01	2	0.51	8	N/A		N/A	

In general, there is a moderate association between the IAA for a relevance aspect and how often it is found in requests. Aspects that are common in a particular domain tend to lead to strong agreement, while rare aspects tend to lead to weak agreement. Across all four domains, there is Pearson correlation of $\rho = 0.44$ between the number of times the annotators assigned an aspect, and their agreement on when it should be assigned. This correlation is strongest in the **Books** domain ($r = 0.50$) and weakest in the **Games** domain ($r = 0.38$).

Table 2. Inter-Annotator Agreement for the Metadata sub-categories across all four domains.

Metadata	Books		Games		Movies		Music	
	IAA	N pos	IAA	N pos	IAA	N pos	IAA	N pos
Audience	0.62	26	1.00	1	N/A		N/A	
Availability	N/A		0.48	11	0.30	5	-0.01	2
Collection & Series	0.75	11	0.36	8	N/A		-0.01	2
Contributor(S)	0.35	21	-0.01	1	1.00	5	0.19	13
Genre	0.58	62	0.56	48	0.56	32	0.82	28
Language	0.48	6	1.00	1	0.65	4	0.82	7
Popularity	-0.01	1	0.79	3	-0.02	2	0.55	5
Properties	0.64	10	0.77	54	0.52	12	-0.01	2
Publisher	0.66	2	-0.02	3	N/A		-0.01	1
Release Date	0.77	33	0.84	39	0.69	41	0.74	17
Soundtrack	N/A		0.85	4	1.00	2	0.71	30
Supplementary Material	N/A		-0.02	3	-0.01	1	0.93	8
Title	0.62	14	0.79	6	0.48	3	0.38	4
Version	-0.01	2	0.66	2	0.48	3	0.57	7

C. Challenges of Collaborative Coding

There are several challenges when annotating complex requests with a complex coding scheme. We find it important to point out that any frequency analyses have to be considered with these caveats in mind.

We have gone through many rounds of discussions of how to interpret the relevance aspects in our coding scheme, with subsequent revisions of the aspects and re-coding of the requests. But still, as will be shown in the next sub-section, Inter-Annotator Agreement (IAA) is far from perfect.

Disagreement in interpretations has consequences for system design. We will never have complete agreement, which prompts the question whether these complex information needs can ever be completely tackled. On the forums, a lot of requests are followed by questions for clarification or specification,

Table 3. Inter-Annotator Agreement for the Experience sub-categories across all four domains.

Experience	Books		Games		Movies		Music	
	IAA	N pos	IAA	N pos	IAA	N pos	IAA	N pos
(Re)Play Value	-0.01	1	-0.01	2	N/A		N/A	
Accessibility	0.37	8	0.49	3	N/A		N/A	
Comprehensiveness	0.54	10	N/A		N/A		N/A	
Impact	0.55	5	N/A		N/A		N/A	
Mood	0.32	9	0.36	8	-0.01	1	0.58	9
Novelty	-0.01	2	N/A		N/A		N/A	
Perspective	0.74	5	0.41	18	-0.02	2	N/A	

Table 4. Inter-Annotator Agreement for the Interactivity sub-categories in the games domain.

Interactivity	Games	
	IAA	N pos
Connectivity	0.25	6
Controls	-0.02	3
Expandability	N/A	
Game Mode	0.70	18

Table 5. Inter-Annotator Agreement for the Search Process sub-categories across all four domains.

Search Process	Books		Games		Movies		Music	
	IAA	N pos	IAA	N pos	IAA	N pos	IAA	N pos
Link To External Resource	-0.01	2	0.85	4	0.66	2	0.97	36
Not This One	-0.03	5	0.36	22	0.64	6	0.49	3
Search History	0.38	4	-0.02	3	-0.03	3	0.48	18
Situation Of Exposure	0.65	6	-0.05	8	0.28	13	0.79	28

showing that human mediators struggle with this. Of course, systems can also ask clarification questions, so low agreement on whether a request contains a specific relevance aspect does not mean that it cannot be usefully incorporated in a search interface and retrieval model.

During the discussions, we noticed that one cause of differences is a difference in domain expertise, with some of us being confident in how to interpret the terminology used and examples given in requests, while others had to frequently look up terms and examples, and remaining in doubt about how to code the request. The **Books** and **Movies** domains were perceived as probably the least challenging, while for the **Games** domain, multiple annotators indicated that they struggled to interpret requests.

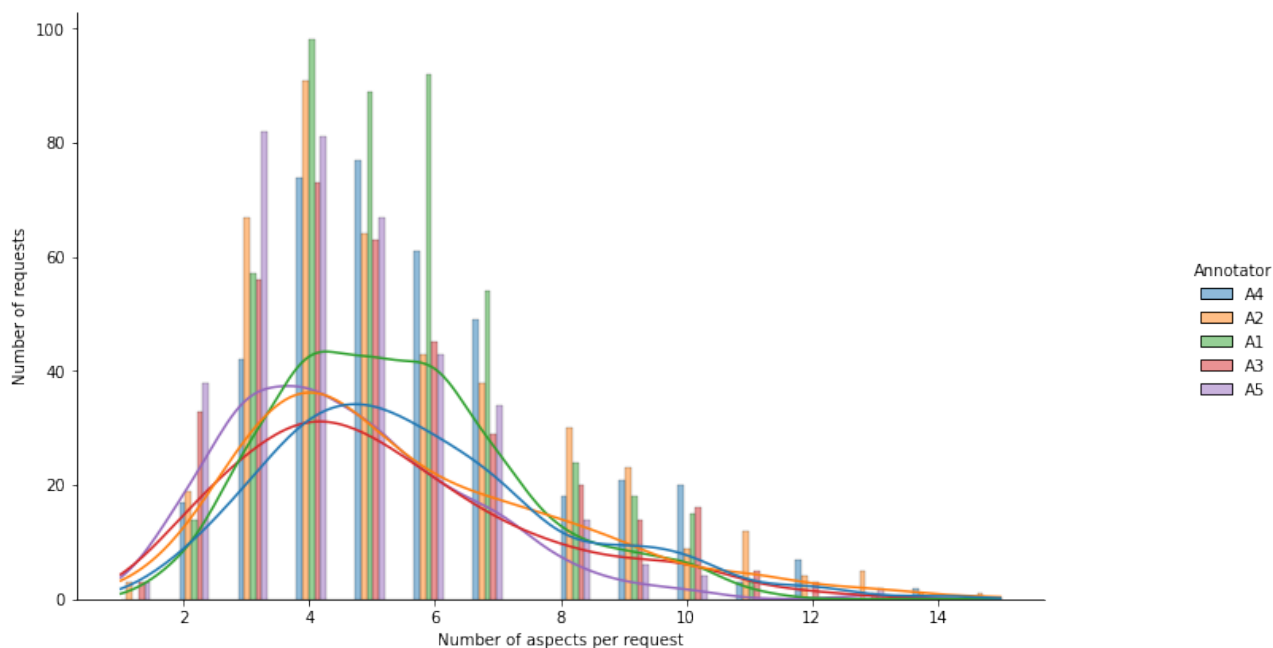
Another challenge is information in a request that fits intuitively in two categories, but through reflection on how it is used, an annotator might put it in only one. An example is a searcher looking for a song they know, of which they only remember that in the music video, the singer had long hair. This describes the video, which is supplementary material to help identify the song, but also describes the singer, which is metadata regarding one of the contributors. Should both aspects be coded? Which of these aspects could help more finding the right song? Another example is a person trying to discover new music that is similar to a known song and in the requests mentions both the artist and title of the song. In a search engine or recommender system, the musical content of the song can be used as input feature, but the artist and title could also be used as features, but the searcher probably does not care who the artist is nor what the title of the song is, but that the music has something they like and want to find other music that is similar.

In summary, the interpretation of nuances might lead to different annotations and priorities.

D. Comparing annotator behaviour

Annotators assigned between 1 and 15 aspects per request. The distribution of the number of aspects per request roughly follows a Poisson distribution (see Figure 1), with a mean of $\lambda = 5.3$ aspects per request. Note that annotator A5 tends to assign fewer aspects than the other annotators.

Figure 1. Distribution of the number of relevance aspects per request, both as histogram and kernel density plot.



The five annotators largely agree on how frequently each relevance aspect occurs. That is, when one annotator assigns an aspect to the majority of requests, the other annotators do so too. For instance, the annotators assign a **Content** aspect to between 74% and 81% of requests. **Language** is rarely assigned by all annotators, ranging between 3% and 6% of requests. For 34 aspects, the maximum difference is below 5 percentage points, for another 17 aspects the maximum difference is between 5 and 10 percentage points. There are two aspects where the difference in frequency is 20 percentage points or more: **Character(s)** (18% and 38%) and **Setting** (9% and 30%).

In the **Movies** domain, the difference for the **Character(s)** aspect is most pronounced, with one annotator assigning to 78% of requests and another to only 33% of requests. This partly explains why agreement for **Character(s)** is lower for **Movies** than for **Books** and **Games** (see Table ??).

We analyse whether there are statistically significant differences between the five annotators in terms of how often they assigned labels from each of the main categories, using a one-way ANOVA and Tukey's Honest Significant Difference (HSD) test with $\alpha = 0.05$.

For the total number of categories assigned, there are statistically significant differences ($p = 0.000$), and the Tukey HSD test indicates that one of the annotators differs significantly from all other annotators.

Zooming in on the six main aspects, we find no difference between annotators for **Context** ($p = 0.284$), **Interactivity** ($p = 0.245$) and **Search process** ($p = 0.316$), partly because these main categories are relatively rare according to all annotators and they therefore mostly assign zero labels per request.

For **Content** ($p = 0.000$), **Metadata** ($p = 0.011$) and **Experience** ($p = 0.000$) there are significant differences. The Tukey HSD tests reveal that of the 10 different annotator pairs, 6 show a statistically significant difference for **Content**, 3 for **Metadata** and 4 for **Experience**.

A closer look into the four domains, significant differences in numbers of assigned categories between annotators in all four domains are observed. The **Books** domain shows significant differences among annotators for all six main categories, while for the other three domains the differences are significant for only three out of six main categories. Experience is the category with significant differences in all four domains, while for **Metadata** this is only the case in the **Books** domain.

What does this mean? Part of the disagreement is explained by the fact that one of the annotators was more strict and assigned fewer aspects than the other annotators. This could mean that some annotators over-specified aspects of an information need (using more aspects than is useful) or that others under-specified aspects (using fewer aspects than is useful). Using these assigned relevance aspects as part of an IR test collection would mean that certain aspects should be left out because of low agreement. But for analysing and understanding casual leisure needs, the low agreement is not necessarily a problem. What matters is that our coding scheme and annotations reveal that many different aspects are considered by searchers, but with different relative frequencies. For informing system design and re-thinking the kinds of aspects that search interfaces could incorporate in terms of e.g. search facets, the relative frequency with which these aspects are observed could be used to prioritise more common aspects over more rarely observed ones, and those with high agreement over those with low agreement.

