Exploiting Distributional Semantics Models for Natural Language Context-aware Justifications for Recommender Systems

Giuseppe Spillo (*), Cataldo Musto, Marco de Gemmis, Pasquale Lops, Giovanni Semeraro
University of Bari - Dip. di Informatica

(*), Bachelor Student - email: giuseppe.spillo@studenti.uniba.it - others: name.surname@uniba.it

Abstract

In this paper we present a methodology to generate context-aware natural language justifications supporting the suggestions produced by a recommendation algorithm. Our approach relies on a natural language processing pipeline that exploits distributional semantics models to identify the most relevant aspects for each different context of consumption of the item. Next, these aspects are used to identify the most suitable pieces of information to be combined in a natural language justification. As information source, we used a corpus of reviews. Accordingly, our justifications are based on a combination of reviews’ excerpts that discuss the aspects that are particularly relevant for a certain context.

In the experimental evaluation, we carried out a user study in the movies domain in order to investigate the validity of the idea of adapting the justifications to the different contexts of usage. As shown by the results, all these claims were supported by the data we collected.

1 Introduction

Recommender Systems (RSs) (Resnick and Varian, 1997) are now recognised as a very effective mean to support the users in decision-making tasks (Ricci et al., 2015). However, as the importance of such technology in our everyday lives grows, it is fundamental that these algorithms support each suggestion through a justification that allows the user to understand the internal mechanisms of the recommendation process and to more easily discern among the available alternatives.

To this end, several attempts have been recently devoted to investigate how to introduce explanation facilities in RSs (Nunes and Jannach, 2017) and to identify the most suitable explanation styles (Gedikli et al., 2014). Despite such a huge research effort, none of the methodologies currently presented in literature diversifies the justifications based on the different contextual situations in which the item will be consumed. This is a clear issue, since context plays a key role in every decision-making task, and RSs are no exception. Indeed, as the mood or the company (friends, family, children) can direct the choice of the movie to be watched, so a justification that aims to convince a user to enjoy a recommendation should contain different concepts depending on whether the user is planning to watch a movie with her friends or with her children.

In this paper we fill in this gap by proposing an approach to generate a context-aware justification that supports a recommendation. Our methodology exploits distributional semantics models (Lenci, 2008) to build a term-context matrix that encodes the importance of terms and concepts in each context of consumption. Such a matrix is used to obtain a vector space representation of each context, which is in turn used to identify the most suitable pieces of information to be combined in a justification. As information source, we used a corpus of reviews. Accordingly, our justifications are based on a combination of reviews’ excerpts that discuss with a positive sentiment the aspects that are particularly relevant for a certain context. Beyond its context-aware nature, another distinctive trait of our methodology is the fact that we generate post-hoc justifications that are completely independent from the underlying recommendation models and completely separated from the step of generating the recommendations.

To sum up, we can summarize the contributions of the article as follows: (i) we propose a method-
ology based on distributional semantics models and natural language processing to automatically learn a vector space representation of the different contexts in which an item can be consumed; (ii) We design a pipeline that exploits distributional semantics models to generate context-aware natural language justifications supporting the suggestions returned by any recommendation algorithm.

The rest of the paper is organized as follows: first, in Section 2 we provide an overview of related work. Next, Section 3 describes the main components of our workflow and Section 4 discusses the outcomes of the experimental evaluation. Finally, conclusions and future work of the current research are provided in Section 5.

2 Related Work

The current research borrows concepts from review-based explanation strategies and distributional semantics models. In the following, we will try to discuss relevant related work and to emphasize the hallmarks of our methodology.

Review-based Explanations. According to the taxonomy discussed in (Friedrich and Zanker, 2011), our approach can be classified as a content-based explanation strategy, since the justifications we generate are based on descriptive features of the item. Early attempts in the area rely on the exploitation of tags (Vig et al., 2009) and features gathered from knowledge graphs (Musto et al., 2016). With respect to classic content-based strategies, the novelty of the current work lies in the use of review data to build a natural language justification. In this research line, (Chen and Wang, 2017) Chen et al. analyze users’ reviews to identify relevant features of the items, which are presented on an explanation interface. Differently from this work, we did not bound on a fixed set of static aspects and we left the explanation algorithm deciding and identifying the most relevant concepts and aspects for each contextual setting. A similar attempt was also proposed in (Chang et al., 2016). Moreover, as previously emphasized, a trait that distinguishes our approach with respect to such literature is the adaptation of the justification based on the different setting in which the item is consumed. The only work exploiting context in the justification process has been proposed by Misztal et al. in (Misztal and Indurkhya, 2015). However, differently from our work, they did not diversify the justifications of the same items on varying of different contextual settings in which the item is consumed, since they just adopt features inspired by context (e.g., “I suggest you this movie since you like this genre in rainy days”) to explain a recommendation. Distributional Semantics Models. Another distinctive trait of the current work is the adoption of distributional semantics models (DMSs) to build a vector space representation of the different contextual situations in which an item can be consumed. Typically, DMSs rely on a term-context matrix, where rows represent the terms in the corpus and columns represents contexts of usage. For the sake of simplicity, we can imagine a context as a fragment of text in which the term appears, as a sentence, a paragraph or a document. Every time a particular term is used in a particular context, such an information is encoded in this matrix. One of the advantages that follows the adoption of DMSs is that they can learn a vector space representation of terms in a totally unsupervised way. These methods, recently inspired methods in the area of word embeddings, such as Word2Vec (Mikolov et al., 2013) and contextual word representations (Smith, 2020). Even if some attempts evaluating RSs based on DMSs already exists (Lops et al., 2009; Musto et al., 2011; Musto et al., 2012; Musto et al., 2014), in our attempt we used DMSs to build a vector-space representation of the different contextual dimensions. Up to our knowledge, the usage of DMSs for justification purposes this is a completely new research direction in the area of explanation.

3 Methodology

Our workflow to generate context-aware justifications based on users’ reviews is shown in Figure 1. In the following, we will describe all the modules that compose the workflow.

Context Learner. The first step is carried out by the Context Learner module, which exploits DMSs to learn a vector space representation of the contexts. Formally, given a set reviews $R$ and a set of $k$ contextual settings $C = \{c_1, \ldots, c_k\}$, this module generates as output a matrix $C_{n,k}$ that encodes the importance of each term $t_i$ in each contextual setting $c_j$. In order to build such a representation, we first split all the reviews $r \in R$ in sentences. Next, let $S$ be the set of previously obtained sentences, we manually annotated a subset of these sentences in order to obtain a
Figure 1: Workflow to generate Context-aware Justifications by Exploiting DSMs

Given such a representation, two different outputs are obtained. First, we can directly extract column vectors \( \hat{c}_j \) from matrix \( C \), which represents the vector space representation of the context \( c_j \) based on DSMs. It should be pointed out that such a representation perfectly fits the principles of DSMs since contexts discussed through the same lemmas will share a very similar vector space representation. Conversely, a poor overlap will result in very different vectors. Moreover, for each column, lemmas may be ranked and those having the highest TF-IDF scores may be extracted. In this way, we obtain a lexicon of lemmas that are relevant for a particular contextual setting, and this can be useful to empirically validate the effectiveness of the approach. In Table 1, we anticipate some details of our experimental session and we report the top-3 lemmas for two different contextual settings starting from a set of movie reviews.

**Ranker.** Given a recommended item (along with its reviews) and given the context in which the item will be consumed (from now on, defined as 'current context'), this module has to identify the most relevant review excerpts to be included in the justification. To this end, we designed a ranking strategy that exploits DSMs and similarity measures in vector spaces to identify suitable excerpts: given a set of \( n \) reviews discussing the item \( i \), \( R_i = \{r_{i,1}, \ldots, r_{i,n}\} \), we first split each \( r_i \) in sentences. Next, we processed the sentences through a sentiment analysis algorithm (Liu, 2012; Petz et al., 2015) in order to filter out those expressing a negative or neutral opinions about the item. The choice is justified by our focus on review excerpts discussing positive characteristics of the item. Next, let \( c_j \) be the current contextual situation (e.g., company=partner), we calculate the cosine similarity between the context vector \( \hat{c}_j \) returned by the CONTEXT LEARNER and a vector...
space representation of each sentence $\tilde{s}_i$. The sentences having the highest cosine similarity w.r.t. to the context of usage $c_j$ are selected as the most suitable excerpts and are passed to the Generator.

**Generator.** Finally, the goal of Generator is to put together the compliant excerpts in a single natural language justification. In particular, we defined a *slot-filling* strategy based on the principles of Natural Language Generation [Reiter and Dale, 1997]. Such a strategy is based on the combination of a *fixed* part, which is common to all the justifications, and a *dynamic* part that depends on the outputs returned by the previous steps. In our case, the top-1 sentence for each current contextual dimension is selected, and the different excerpts are merged by exploiting simple connectives, such as adverbs and conjunctions. An example of the resulting justifications is provided in Table 2.

### Table 1: Top-3 lemmas returned by the CONTEXT LEARNER module for two couples of different contextual settings in the MOVIE domain.

| Unigrams            | Attention=high                                                                 | Attention=low                                                                 |
|---------------------|-------------------------------------------------------------------------------|-------------------------------------------------------------------------------|
| Bigrams             | engaging, attentive, intense                                                  | simple, smooth, easy                                                         |
|                     | intense plot, slow movie, life metaphor                                        | easy vision, simple movie, simple plot                                       |

4 Experimental Evaluation

The experimental evaluation was designed to identify the best-performing configuration of our strategy, on varying of different combinations of the parameters of the workflow (*Research Question 1*), and to assess how our approach performs in comparison to other methods (both context-aware and non-contextual) to generate *post-hoc justifications* (*Research Question 2*). To this end, we designed a user study involving 273 subjects (male=50%, degree or PhD=26.04%, age≥35=49.48%, already used a RS=85.4%) in the movies domain. Interest in movies was indicated as medium or high by 62.78% of the sample. Our sample was obtained through the availability sampling strategy, and it includes students, researchers in the area and people not skilled with computer science and recommender systems. As in [Tintarev and Masthoff, 2012], whose protocol was took as a reference in several subsequent research in the area of explanation (Musto et al., 2019), we evaluated the following metrics: transparency, persuasiveness, engagement and trust through a post-usage questionnaire.

**Experimental Design.** To run the experiment, we deployed a web application [8] implementing the methodology described in Section 3. Next, as a first step, we identified the relevant contextual dimensions for each domain. Contexts were selected by carrying out an analysis of related work of context-aware recommender systems in the MOVIE domain. In total, we defined 3 contextual dimensions, that is to say, mood (great, normal), company (family, friends, partner) and level of attention (high, low). To collect the data necessary to feed our web application, we selected a subset of 300 popular movies (according to IMDB data) discussed in more than 50 reviews in the Amazon Reviews dataset [5]. This choice is motivated by our need of a large set of sentences discussing the item in each contextual setting. These data were processed by exploiting lemmatization, POS-tagging and sentiment analysis algorithms available in CoreNLP [4] and Stanford Sentiment Analysis Algorithm [5] tool. Some statistics about the final dataset are provided in Table 3.

![Figure 2: Building a lemma-context matrix C by exploiting distributional semantics models](image-url)
### Table 2: Context-aware justifications for the MOVIE domain. Automatically extracted review excerpts are reported in italics.

| Movie       | Justification                                                                 |
|-------------|-------------------------------------------------------------------------------|
| Company=Partner | You should watch 'Stranger than Fiction'. It is a good movie to watch with your partner because it has a very romantic end. Moreover, plot is very intense. |
| Company=Friends | You should watch 'Stranger than Fiction'. It is a good movie to watch with friends since the film crackles with laughter and pathos and it is a classy sweet and funny movie. |

In order to compare different configurations of the workflow, we designed several variant obtained by varying the vocabulary of lemmas. In particular, we compared the effectiveness of simple unigrams, of bigrams and their merge. In the first case, we encoded in our matrix just single lemmas (e.g., service, meal, romantic, etc.) while in the second we stored combination of nouns and adjectives (e.g., romantic location). Due to space reasons, we can’t provide more details about the lexicons we learnt, and we suggest to refer again to Table 1 for a qualitative evaluation of some of the resulting representations. Our representations based on DSMs were obtained by starting from a set of 1,905 annotations for the movie domain, annotated by three annotators by adopting a majority vote strategy. To conclude, each user involved in the experiment carried out the following steps:

1. **Training, Context Selection and Generation of the Recommendation.** First, we asked the users to provide some basic demographic data and to indicate their interest in movies. Next, each user indicated the context of consumption of the recommendation, by selecting a context among the different contextual settings we previously indicated (see Figure 3-a). Given the current context, a suitable recommendation was identified and presented to the user. As recommendation algorithm we used a content-based recommendation strategy exploiting users’ reviews.

2. **Generation of the Justification.** Given the recommendation and the current context of consumption, we run our pipeline to generate a context-aware justification of the item adapted to that context. In this case, we designed a between-subject protocol. In particular, each user was randomly assigned to one of the three configurations of our pipeline and the output was presented to the user along with the recommendation (see Figure 3-b).

Clearly, the user was not aware of the specific configuration he was interacting with.

3. **Evaluation through Questionnaires.** Once the justification was shown, we asked the users to fill in a post-usage questionnaire. Each user was asked to evaluate transparency, persuasiveness, engagement and trust of the recommendation process through a five-point scale (1=strongly disagree, 5=strongly agree). The questions the users had to answer follow those proposed in (Tintarev and Masthoff, 2012). Due to space reasons, we can’t report the questions and we suggest to interact with the web application to fill in the missing details.

4. **Comparison to baselines.** Finally, we compared our method to two different baselines in a within-subject experiment. In this case, all the users were provided with two different justifications styles (i.e., our context-aware justifications and a baseline) and we asked the users to choose the one they preferred. As for the baselines, we focused on other methodologies to generate post-hoc justifications and we selected (i) a context-aware strategy to generate justifications, which is based on a set of manually defined relevant terms for each context; (ii) a method to generate non-contextual review-based justifications that relies on the automatic identification of relevant aspects and on the selection of compliant reviews excerpts containing such terms. Such approach partially replicates that presented in (Musto et al., 2020).

**Discussions of the Results** Results of the first experiment, that allows to answer to Research Question 1, are presented in Table 3. The values in the tables represent the average scores provided by the users for each of the previously mentioned questions. As for the movie domain, results show that the overall best results are obtained by us-
Table 3: Statistics of the dataset

|        | #Items | #Reviews | #Sentences | #Positive Sent. | Avg. Sent./Item | Avg. Pos. Sent./Item |
|--------|--------|----------|------------|-----------------|----------------|---------------------|
| MOVIES | 307    | 153,398  | 1,464,593  | 560,817         | 4,770.66       | 1,826.76            |

(a) Context Selection

(b) Recommendation and Justification

Figure 3: Interaction with the web application.

ing a vocabulary based on unigrams and bigrams. This first finding provides us with an interesting outcome, since most of the strategies to generate explanations are currently based on single keywords and aspects. Conversely, our experiment showed that both adjectives as well as couples of co-occurring terms are worth to be encoded, since they catch more fine-grained characteristics of the item that are relevant in a particular contextual setting. Overall, the results we obtained confirmed the validity of the approach. Beyond the increase in transparency, high evaluations were also noted for persuasion and engagement metrics. This outcome confirms how the identification of relevant reviews’ excerpts can lead to satisfying justifications. Indeed, differently from feature-based justifications, that typically rely on very popular and well-known characteristics of the movie, as the actors or the director, more specific aspects of the items emerge from users’ reviews.

Next, in order to answer to Research Question 2, we compared the best-performing configurations emerging from Experiment 1 to two different baselines. The results of these experiments are reported in Table 5, which show the percentage of users who preferred our context-aware methodology based on DSMs to both the baselines. In particular, the first comparison allowed us to assess the effectiveness of a vector space representation of contexts based on DSMs with respect to a simple context-aware justification method based on a fixed lexicon of relevant terms, while the second comparison investigated how valid was the idea of diversifying the justifications based on the different contextual settings in which the items is consumed. As shown in the table, our approach was the preferred one in both the comparisons. It should be pointed out that the gaps are particularly large when our methodology is compared to a non-contextual baseline. In this case, we noted a statistically significant gap (p ≤ 0.05) for all the metrics, with the exception of trust. This suggests that diversifying the justifications based on the context of consumption is particularly appreciated by the users. This confirms the validity of our intuition, which led to a completely new research direction in the area of justifications for recommender systems.

5 Conclusions and Future Work

In this paper we presented a methodology that exploits DSMs to build post-hoc context-aware natural language justifications supporting the suggestions generated by a RS. The hallmark of this work is the diversification of the justifications based on the different contextual settings in which the items will be consumed, which is a new research direction in the area. As shown in our experiments, our justifications were largely preferred by users. This confirms the effectiveness of our approach and paves the way to several future research directions, such as the definition of personalized justi-
### Metrics / Configuration

| Metrics / Configuration | Unigrams | Bigrams | Uni+Bigrams |
|------------------------|----------|---------|-------------|
| Transparency           | 3.38     | 3.81    | 3.64        |
| Persuasion             | 3.36     | 3.62    | 3.54        |
| Engagement             | 3.54     | 3.72    | 3.70        |
| Trust                  | 3.44     | 3.66    | 3.61        |

Table 4: Results of Experiment 1 for the MOVIE domain. The best-performing configuration is reported in **bold** and underlined.

### Metrics / Choice vs. Context-aware Static Baseline vs. Non-Contextual Baseline

| Metrics / Choice | CA+DSMs | Baseline | Indifferent | CA+DSMs | Baseline | Indifferent |
|------------------|---------|----------|-------------|---------|----------|-------------|
| Transparency     | 52.38%  | 38.10%   | 19.52%      | 53.21%  | 34.47%   | 12.32%      |
| Persuasion       | 54.10%  | 36.33%   | 19.57%      | 55.17%  | 32.33%   | 12.50%      |
| Engagement       | 49.31%  | 39.23%   | 11.56%      | 44.51%  | 32.75%   | 22.74%      |
| Trust            | 42.86%  | 39.31%   | 17.83%      | 42.90%  | 42.11%   | 14.99%      |

Table 5: Results of Experiment 2, comparing our approach (CA+DSMs) to a context-aware baseline that does not exploit DSMs (CA Static) and to a non-contextual baseline that exploit users’ reviews (review-based). The configuration preferred by the higher percentage of users is reported in **bold**.

...fication as well as the generation of hybrid justifications that combine elements gathered from user-generated content (as the reviews) with descriptive characteristics of the items. Finally, we will also evaluate the use of ontologies and rules ([Laera et al., 2004](#)) in order to implement reasoning mechanisms to better identify the most relevant aspects in the reviews.

### References

- [Chang et al.2016](#) Shuo Chang, F Maxwell Harper, and Loren Gilbert Terveen. 2016. Crowd-based Personalized Natural Language Explanations for Recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pages 175–182. ACM.
- [Chen and Wang2017](#) Li Chen and Feng Wang. 2017. Explaining Recommendations based on Feature Sentiments in Product Reviews. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, pages 17–28. ACM.
- [Friedrich and Zanker2011](#) Gerhard Friedrich and Markus Zanker. 2011. A taxonomy for generating explanations in recommender systems. *AI Magazine*, 32(3):90–98.
- [Gedikli et al.2014](#) Fatih Gedikli, Dietmar Jannach, and Mouzhi Ge. 2014. How should I explain? a comparison of different explanation types for recommender systems. *International Journal of Human-Computer Studies*, 72(4):367–382.
- [Laera et al.2004](#) Loredana Laera, Valentina Tamma, Trevor Bench-Capon, and Giovanni Semeraro. 2004. Sweetprolog: A system to integrate ontologies and rules. In *International Workshop on Rules and Rule Markup Languages for the Semantic Web*, pages 188–193. Springer.
- [Lenci2008](#) Alessandro Lenci. 2008. Distributional semantics in linguistic and cognitive research. *Italian Journal of Linguistics*, 20(1):1–31.
- [Liu2012](#) Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.
- [Lops et al.2009](#) Pasquale Lops, Marco de Gemmis, Giovanni Semeraro, Cataldo Musto, Fedelucio Narducci, and Massimo Bux. 2009. A semantic content-based recommender system integrating folksonomies for personalized access. In *Web Personalization in Intelligent Environments*, pages 27–47. Springer.
- [Manning et al.1999](#) Christopher D Manning, Christopher D Manning, and Hinrich Schütze. 1999. *Foundations of statistical natural language processing*. MIT press.
- [Mikolov et al.2013](#) Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.
- [Misztal and Indurkhya2015](#) Joanna Misztal and Bipin Indurkhya. 2015. Explaining contextual recommendations: Interaction design study and prototype implementation. In *IntRS@RecSys*, pages 13–20.
- [Musto et al.2011](#) C. Musto, G. Semeraro, P. Lops, and M. de Gemmis. 2011. Random indexing and negative user preferences for enhancing content-based
Recommender Systems. In EC-Web 2011, volume 85 of Lecture Notes in Business Inf. Processing, pages 270–281. Springer.

[Musto et al.2012] C. Musto, F. Narducci, P. Lops, G. Semeraro, M. De Gemmis, M. Barbieri, J. Koirst, V. Pronk, and R. Clout. 2012. Enhanced semantic tv-show representation for personalized electronic program guides. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 7379 LNCS:188–199. cited By 19.

[Musto et al.2014] C. Musto, G. Semeraro, P. Lops, and M. de Gemmis. 2014. Combining distributional semantics and entity linking for context-aware content-based recommendation. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 8538:381–392. cited By 19.

[Musto et al.2016] Cataldo Musto, Fedelucio Narducci, Pasquale Lops, Marco De Gemmis, and Giovanni Semeraro. 2016. Explod: A framework for explaining recommendations based on the linked open data cloud. In Proceedings of the 10th ACM Conference on Recommender Systems, RecSys ’16, pages 151–154, New York, NY, USA. ACM.

[Musto et al.2019] Cataldo Musto, Pasquale Lops, Marco de Gemmis, and Giovanni Semeraro. 2019. Justifying recommendations through aspect-based sentiment analysis of users reviews. In Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization, pages 4–12.

[Musto et al.2020] Cataldo Musto, Marco de Gemmis, Pasquale Lops, and Giovanni Semeraro. 2020. Generating post hoc review-based natural language justifications for recommender systems. User Modeling and User-Adapted Interaction, pages 1–45.

[Nakagawa and Mori2002] Hiroshi Nakagawa and Tatsumori Mori. 2002. A simple but powerful automatic term extraction method. In COLING-02 on COMPUTERM 2002: second international workshop on computational terminology-Volume 14, pages 1–7. Association for Computational Linguistics.

[Nunes and Jannach2017] Ingrid Nunes and Dietmar Jannach. 2017. A systematic review and taxonomy of explanations in decision support and recommender systems. User Modeling and User-Adapted Interaction, 27(3-5):393–444.

[Petz et al.2015] Gerald Petz, Michal Karpowicz, Harald Fürschuß, Andreas Auinger, Václav Stříteský, and Andreas Holzinger. 2015. Reprint of: Computational approaches for mining user’s opinions on the web 2.0. Information Processing & Management, 51(4):510–519.

[Reiter and Dale1997] Ehud Reiter and Robert Dale. 1997. Building applied natural language generation systems. Natural Language Engineering, 3(1):57–87.

[Resnick and Varian1997] Paul Resnick and Hal R Varian. 1997. Recommender systems. Communications of the ACM, 40(3):56–58.

[Ricci et al.2015] Francesco Ricci, Lior Rokach, and Bracha Shapira. 2015. Recommender systems: introduction and challenges. In Recommender systems handbook, pages 1–34. Springer.

[Smith2020] Noah A Smith. 2020. Contextual word representations: putting words into computers. Communications of the ACM, 63(6):66–74.

[Tintarev and Masthoff2012] Nava Tintarev and Judith Masthoff. 2012. Evaluating the Effectiveness of Explanations for Recommender Systems. UMUAI, 22(4-5):399–439.

[Vig et al.2009] Jesse Vig, Shilad Sen, and John Riedl. 2009. Tagsplanations: explaining recommendations using tags. In Proceedings of the 14th international conference on Intelligent user interfaces, pages 47–56. ACM.