Improving the Relevance of Product Search for Queries with Negations

Felice Antonio Merra
felmerra@amazon.com
Amazon, Inc.
Berlin, Germany

Omar Zaidan
ozaidan@amazon.com
Amazon, Inc.
Berlin, Germany

Fabricio de Sousa Nascimento
fabun@amazon.com
Amazon, Inc.
Tokyo, Japan

ABSTRACT

Product search engines (PSEs) play an essential role in retail websites as they make it easier for users to retrieve relevant products within large catalogs. Despite the continuous progress that has led to increasingly accurate search engines, a limited focus has been given to their performance on queries with negations. Indeed, while we would expect to retrieve different products for the queries "iPhone 13 cover with ring" and "iPhone 13 cover **without** ring", this does not happen in popular PSEs with the latter query containing results with the unwanted ring component. The limitation of modern PSEs in understanding negations motivates the need for further investigation.

In this work, we start by defining the negation intent in users queries. Then, we design a transformer-based model, named Negation Detector for Queries (ND4Q), that reaches optimal performance in negation detection (+95% on accuracy metrics). Finally, having built the first negation detector for product search queries, we propose a negation-aware filtering strategy, named Filtering Irrelevant Products (FIP). The promising experimental results in improve the PSE relevance performance using *FIP* (+9.41% on *nDCG*@16 for queries where the negation starts with "*without*") pave the way to additional research effort towards negation-aware PSEs.

CCS CONCEPTS

• **Information systems** → **Retrieval models and ranking**; **Query intent**.

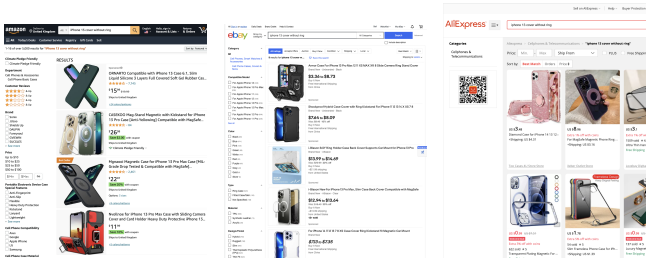
KEYWORDS

Product Search, Natural Language Processing, Negation

ACM Reference Format:
Felice Antonio Merra, Omar Zaidan, and Fabricio de Sousa Nascimento. 2023. Improving the Relevance of Product Search for Queries with Negations. In *Companion Proceedings of the ACM Web Conference 2023 (WWW '23 Companion), April 30-May 4, 2023, Austin, TX, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3543873.3587319

1 INTRODUCTION

Retail stores, such as Amazon, eBay, and AliExpress, are popular choices used by people to search for what they want to buy and, if satisfied, purchase. To guarantee users satisfaction, online retailers rely on product search engines (PSEs) that, given a user search query, have to retrieve a few dozens relevant products to fulfill the user's request [22, 23]. Providing high-quality results has attracted more and more attention from both academia and industry. A huge research effort has been spent on using advanced natural language processing (NLP) techniques on query, e.g., intent classification [1] and query rewriting [13], and product, e.g., summarization [24] and attribute values extractions [26], in order to improve the PSE's performance. However, a limited attention has been paid to understanding queries with negations such that the PSE can return relevant results. Indeed, as shown in Figure 1, popular PSEs might fail to handle this class of queries by returning products clearly violating the negation. For instance, many products retrieved by the websites of Figure 1 contain the unwanted ring component specified in the query "iPhone 13 cover without ring".



(a) Amazon.com (b) eBay.com (c) AliExpress.com

Figure 1: Examples of irrelevant results retrieved by the PSEs of three online retailers for the query "iPhone 13 cover without ring".

In this work, we start by defining the negation in users' queries. This definition is motivated by the fact that the presence of a negation cue like "no" in a user query is not enough to recognize if the query has a *negation intent*, i.e., the user wants to remove a product property. Then, we propose a novel transformer-based negation detector, named Negation Detector for Queries (ND4Q), that, reaching optimal performance in detecting negations, contributes to improve the relevance of retrieved products for the queries on which ND4Q has detected a negation.

To sum up, the main contributions of this work are as follows:

- we define the negation in user query providing examples of queries with and without a negation intent;
- we design and test a transformer-based negation detector, ND4Q, that was trained on a dataset of human-annotated

WWW '23 Companion, April 30-May 4, 2023, Austin, TX, USA

ACM ISBN 978-1-4503-9419-2/23/04.
https://doi.org/10.1145/3543873.3587319

queries such that it recognized negation based on the previous definition;
- we propose an algorithm, named Filtering Irrelevant Products (FIP), to improve the relevance of the retrieved products for the queries with negations.

We perform extensive experiments to demonstrate the high quality of ND4Q in detecting the negation in users' queries and FIP in improving the relevance of the retrieved products on the queries with negations. This work paves the way for novel research effort for improving PSEs on queries with negations.

2 RELATED WORK

Processing negation in NLP applications is non-trivial. Most of the research effort has been focused on developing models for negation detection in the medical domain. In that context, accurate detection is crucial since patients' health is at stake [3, 4]. As common in any NLP task, the negation detectors have evolved from rule-based algorithms, e.g., NegEx [4], ConTex [25], pyContext [3], and DEEPEN [15], to machine learning (ML) methods, e.g., Support Vector Machine (SVM) [7] and Conditional Random Field (CRF) [5]. With the advent of deep learning, novel solutions have rapidly appeared being more and more able to capture negations. Qian et al. [18] and Lazib et al. [11] designed the first CNN and RNN detectors, respectively, that outperformed all the previous rule-based and ML models. More recently, with transformers [6], novel detectors outperformed the previous models in widely tested corpora, e.g., NegBERT [9]. Indeed, we built our detector as a transformer model that differs from NegBERT. While NegBERT approaches the negation detection as a 2-stage task by firstly detecting the negation cue, and then, the negation scope only on the sentences where it found a cue; our model, i.e., ND4Q, detects both cues and scopes within a unique sequence labelling task. The main motivation is that ND4Q is used on users queries which are much shorter than periods processed by NegBERT, e.g., Sherlock Dataset [16].

While the research effort has been mainly focused on having more and more precise models in detecting the negation on a few domain-specific corpora, less attention has been payed to negation usage on PSEs. The most similar studies to our work have focused on showing that negations can harm clinical search systems [10, 12]. To address these issues, both filtering and indexing solutions have been successfully proposed to reduce the weak behavior of modern PSEs on queries with negations [2, 10, 12]. That said, this work aims to explore how to detect negations in users' queries as well as investigate filtering-based solutions to improve the PSE performance on this class of queries.

3 DEFINITIONS

A PSE returns a list of products sorted by relevance for each query. Commonly, a query q is a sequence of words on which natural language processing (NLP) techniques are used to extract tokens (or embeddings) that will be used to retrieve the most relevant products based on the tokens (or embeddings) extracted from each product p textual data, e.g., titles and descriptions. We define $g(q, p)$ the relevance function that produces a score based on keywords similarities, e.g., BM25 [20], or embeddings (semantic) similarities, e.g., DSSM [17].

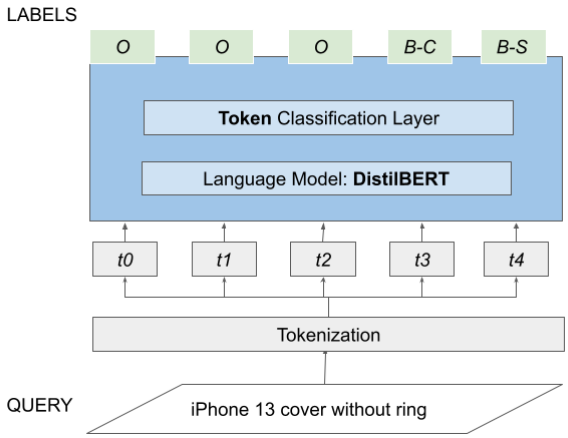


Figure 2: Architecture of ND4Q.

Since $g(\cdot)$ depends on the text of q and p, the PSE performance relies on its robustness against a complex linguistic phenomenon such as the negation. Definition 3.1 presents our proposed definition of negations in users' queries.

Definition 3.1 (Negation in User Query). *Let q be a user query and t_i be the i^{th} token in q, we define the negation cue (NC_q) as the set of tokens expressing a negation, and the negation scope (NS_q) as the set tokens affected by the negation, if the negation is used by the user* ***to search for a product without one or more properties****. In this case, we say that q has a* ***negation intent*** *and $N_q = (NC_q, NS_q)$ is the* ***Negation in User Query****.*

Example 1. For example, in the query

`iPhone 13 cover` **`without`** `ring`

$NC_q = [t_3] =$ ["*without*"] is the negation cue and $NS_q = [t_4] =$ ["*ring*"] is the negation scope because the users wants a product without a property, e.g., a cover without the ring component.

Example 2. Differently from Example 1, in the query

`Spiderman No Way Home`

even if $t_1 =$ "*no*" could be a negation cue, it does not form with "*way home*" a **Negation in User Query** since there is not a negation intent, i.e., the user is not using the token "*no*" to remove a property from the requested product.

4 METHODOLOGY

4.1 Detecting Negation

We formulate the negation detection problem as a sequence labelling [8] task since the identification of negation cues and scopes involves the algorithmic assignment of a categorical label to each token of a sequence of tokens (the query).

4.1.1 Architecture. Figure 2 shows the architecture of the transformer based negation detector, named Negation Detector for Queries (ND4Q), designed and tested in this work. As illustrated in Figure 2, after the tokenization of the input query q, ND4Q processes the list of tokens via a pre-trained language model, whose output is a matrix of query contextualized token embeddings where H is the dimension of each embedding. This matrix is then processed via

a feed-forward linear layer that maps each H-dimensional token embedding into an L-dimensional one, where L is the number of possible labels. Finally, for each token ND4Q associates the label with the highest value.

4.1.2 Implementation details. We use DistilBERT [21] as the pre-trained transformer model since it is optimal for real-world applications. Indeed, it is 40% smaller, 60% faster, and only 5% less precise than BERT [6]. As a consequence, we perform the query tokenization with the `distilbert-base-uncased` tokenizer available in HuggingFace[1]. We set $H = 768$ since we stack a linear classification layer on the top of the last DistilBERT transformer block, and $L = 9$ since it is the number of Beginning-Inside-Outside [19] labels chosen to capture the negation. In particular, the set of labels is $\mathcal{L} := \{0 : \text{'B-C'}, 1 : \text{'I-C'}, 2 : \text{'B-S'}, 3 : \text{'I-S'}, 4 : \text{'B-SC'}, 5 : \text{'I-SC'}, 6 : \text{'B-CS'}, 7 : \text{'I-CS'}, 8 : \text{'O'}\}$, where 'CS' and 'SC' are used when the negation cue is either the prefix, e.g., "***anti***coagulant", or the suffix, e.g., "wire***less***", of the negation scope. The label 'O' is associated with the tokens not affected by the negation. For instance,

$$\underset{\text{O}}{\texttt{iPhone}}\ \underset{\text{O}}{\texttt{13}}\ \underset{\text{O}}{\texttt{cover}}\ \underset{\text{B-C}}{\textbf{without}}\ \underbrace{\texttt{ring}}_{\text{B-S}}$$

We treat this task as a multi-class classification problem, and we train the entire model by minimizing a multi-class cross entropy.

4.2 Improving Product Search

We design the Filtering Irrelevant Products (FIP) algorithm based on the intuition that if there is a negation intent in a query, then we can improve the product relevance by removing all the products in which the negated scope is not close to a negation cue. First, we build a list of possible negation cues extracted from the ND4Q training set of queries (seeSection 5.1.1). For instance, "*no*", "*without*", "*free*", and "*less*" are some of the negation cues within this set. Then, we process a given list of retrieved products by removing the ones where the negation scope of the query is in the text of a product, e.g., title, but it is not close to one of the selected negation cues.

Example 3. Suppose we use FIP on the query "iPhone 13 cover without ring" where $N = (["without"], ["ring"])$. If the title of a retrieved product is

`Red ring`**`less`**` cover for iPhone 13`

then FIP maintains the product. In contrast, if the title is

`White metal cover with ring`

then FIP removes the product because "*ring*" is not next to a negation cue.

5 EXPERIMENTAL RESULTS

In this section, we first present the experiments on the negation detector model, then we present and discuss the results of the proposed filtering technique.

5.1 ND4Q Experiments

5.1.1 Dataset. We train, validate, and test ND4Q on a dataset of about 10 thousand English queries with negations annotated by humans. The dataset statistics and splitting details are reported in Table 1. We used a regular expression to collect a pool of queries containing a possible negation cue such as "*no*" and "*without*" from a real-world dataset sampled from the logs of a retailer site. Then, within this set of queries, we have randomly sampled the ones on which we asked annotator to select the negation cue and scope. If at least two annotators out of three agreed, then we have associated their annotations to the query, otherwise we have removed that query from the dataset. At the end of this procedure, we have cleared away about the 10% of the annotated queries.

[1]https://huggingface.co/

Table 1: Statistics of the annotated queries.

Statistics	**Train** [80%]	**Valid** [10%]	**Test** [10%]
Annotated Queries	7,714	858	953
Q. with negations	4,131 (53.6%)	465 (54.2%)	497 (52.2%)

Table 2: Performance on the negation detection task.

Detector	*Acc*	P_{NC}	R_{NC}	P_{NS}	R_{NS}
Memorized Neg.	85.10%	76.21%	74.41%	81.91%	67.54%
ND4Q	**95.38%**	**92.41%**	**98.10%**	**91.18%**	**95.18%**

5.1.2 Evaluation Metrics. We evaluate ND4Q measuring the fraction of the queries in the test set that gets labeled correctly in their entirety (*Acc*), and both the precision (*P*) and recall (*R*) on the detection of the negation cues (*NC*) and negation scopes (*NS*).

5.1.3 Baseline. We compare ND4Q with a data memorization baseline, named *Memorized Negations*. This model stores the negations detected in the training data into a dictionary that is used to detect the negations in the test set. For instance, if the query "iPhone 13 cover without ring" is in the training data, then, *Memorized Neg.* will detect "*without ring*" as a negation in the test query "S21 Samsung case without ring" since it has been recorded from the training set.

5.1.4 Results. Table 2 reports the accuracy performance measured on the test set. The first observation is that our solution is more accurate than the baseline model in both entirely annotating queries 95.38% vs. 85.10% and correctly recognizing the entire negation spans, e.g., P_{NC} and P_{NS} are higher than 91%. These results are coherent with the recent findings in [9], where a BERT-based negation detector outperforms baseline models on several corpora. Having established the high quality of the detector's predictions, we experiment below the performance of FIP on search results.

5.2 FIP Experiments

5.2.1 Experimental Settings. We test FIP on the ranked list of a baseline model applied on 1,000 queries randomly sampled from a real-world dataset retrieved from the logs of a PSE. For each query-product pair, we have collected human binary relevance feedback that are later used to evaluate the top-8 and top-16 ranking performance with Precision (*Pr*) and normalized Discounted Cumulative Gain (*nDCG*) [14]. We report the results relative variation $\Delta_{\%}$ between the proposed model and the baseline in Table 3.

5.2.2 Results and Discussion. Analyzing Table 3, we observe that the baseline model and FIP have similar performance on the full set of queries with negation. Since negations can be expressed with different negation cues, we hypothesize that the retrieval performance might depend on the type of negations. In Table 3,

Table 3: $\Delta_{\%}$ results between FIP and the baseline model.

	Perc.	*Pr*@8	*Pr*@16	*nDCG*@8	*nDCG*@16
Full Set.	100%	-0.21%	+0.06%	-0.25%	-0.03%
Analysis on different negation cues.					
Neg. cue "*less*"	57.3%	+0.05%	+0.39%	+0.05%	+0.28%
Neg. cue "*no*"	21.3%	-0.08%	+0.16%	-0.40%	-0.12%
Neg. cue "*free*"	13.3%	-0.23%	-0.82%	-0.07%	-0.41%
Neg. cue "*anti*"	3.3%	-1.63%	-3.09%	-2.10%	-2.78%
Neg. cue "*low*"	1.5%	-2.02%	+0.53%	-1.30%	+0.19%
Neg. cue "*without*"	1.4%	0.00%	+14.29%	+1.01%	+9.41%
Neg. cue "*zero*"	1.1%	-13.64%	-14.95%	-12.52%	-13.62%
Others	0.8%	+1.89%	-2.94%	+1.32%	-1.61%

we report the values for the eight most frequent negation cues in the test set. We start the analysis for the most popular negation cue, i.e., "*less*", that appears in 57.3% of the queries. We observe that FIP can slightly improve the baseline model, i.e., +0.28% on *nDCG*@16. Then, we verify that the queries where the negation cues are "*no*", "*free*", "*anti*", "*zero*", and "*low*" do not show a clear trend since FIP can improve the baseline as well as worsen it. We explain this mixed performance with the fact that these cues, even if recognized a *Negation in User Query* by ND4Q, are not always used with a negation intent making the filtering procedure more difficult. For instance, the query "anti-theft envelope" has a different meaning from "envelope without theft", that could be the title of a product considered relevant from FIP. It follows that additional studies could be focused on improving ND4Q in not recognizing these cases as negations. Finally, we focus on the queries where the negation cue is "*without*". The results show that FIP outperforms the baseline model of +9.41% on *nDCG*@16 and +14.29% on *Pr*@16. We explain this behavior by the fact that "*without*" is not commonly used by users (1.4% of queries with negations). As a consequence, since sellers use more popular cues, e.g., "*free*" and "*less*", then a standard PSE will suffer more than a PSE implementing FIP in retrieving relevant products for the queries containing "*without*". In fact, unlike the baseline, our solution favors the retrieval of products that respect the negation regardless of the negation cue.

From the analysis of the results, we can conclude that, while our solution improves the performance of a PSE for 60% of the queries where the negation cues are "*without*" and "*less*", further work is needed to make it consistently more accurate on the other cases.

6 CONCLUSION

This work has investigated the impact of negations on product search engines. Firstly, we have proposed the definition of negations in users' queries, then, we have designed a transformer-based negation detector (ND4Q) that has shown optimal accuracy performance (i.e., accuracy greater than 95%). Having a model capable of detecting negations, we have designed a filtering solution (FIP) to improve the search relevance. The experimental results have shown that FIP improves a baseline PSE by more than 10% on queries containing negations cued by "without". We believe that this work lays the groundwork for new research directions on PSE that can properly capture and handle queries with negations.

REFERENCES

[1] Ali Ahmadvand, Surya Kallumadi, Faizan Javed, and Eugene Agichtein. 2020. JointMap: Joint Query Intent Understanding For Modeling Intent Hierarchies in E-commerce Search. In *SIGIR*. ACM, 1509–1512.

[2] Mordechai Averbuch, Tom H. Karson, Benjamin Ben-Ami, Oded Maimon, and Lior Rokach. 2004. Context-Sensitive Medical Information Retrieval. In *MedInfo (Studies in Health Technology and Informatics, Vol. 107)*. IOS Press, 282–286.

[3] Brian E. Chapman, Sean Lee, Hyunseok Peter Kang, and Wendy Webber Chapman. 2011. Document-level classification of CT pulmonary angiography reports based on an extension of the ConText algorithm. *J. Biomed. Inf.* 44, 5 (2011), 728–737.

[4] Wendy Webber Chapman, Will Bridewell, Paul Hanbury, Gregory F. Cooper, and Bruce G. Buchanan. 2001. A Simple Algorithm for Identifying Negated Findings and Diseases in Discharge Summaries. *J. Biomed. Inf.* 34, 5 (2001), 301–310.

[5] Isaac G. Councill, Ryan T. McDonald, and Leonid Velikovich. 2010. What's great and what's not: learning to classify the scope of negation for improved sentiment analysis. In *NeSp-NLP@ACL*. University of Antwerp, 51–59.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT (1)*. Association for Computational Linguistics, 4171–4186.

[7] Noa P. Cruz Díaz, Maite Taboada, and Ruslan Mitkov. 2016. A machine-learning approach to negation and speculation detection for sentiment analysis. *J. Assoc. Inf. Sci. Technol.* 67, 9 (2016), 2118–2136.

[8] Alex Graves. 2012. *Supervised Sequence Labelling with Recurrent Neural Networks*. Studies in Computational Intelligence, Vol. 385. Springer.

[9] Aditya Khandelwal and Suraj Sawant. 2020. NegBERT: A Transfer Learning Approach for Negation Detection and Scope Resolution. In *LREC*. European Language Resources Association, 5739–5748.

[10] Bevan Koopman and Guido Zuccon. 2014. Understanding negation and family history to improve clinical information retrieval. In *SIGIR*. ACM, 971–974.

[11] Lydia Lazib, Yanyan Zhao, Bing Qin, and Ting Liu. 2016. Negation Scope Detection with Recurrent Neural Networks Models in Review Texts. In *ICYCSEE (1)*, Vol. 623. Springer, 494–508.

[12] Nut Limsopatham, Craig Macdonald, Richard McCreadie, and Iadh Ounis. 2012. Exploiting term dependence while handling negation in medical search. In *SIGIR*. ACM, 1065–1066.

[13] Aritra Mandal, Ishita K. Khan, and Prathyusha Senthil Kumar. 2019. Query Rewriting using Automatic Synonym Extraction for E-commerce Search. In *eCOM@SIGIR (CEUR Workshop Proceedings, Vol. 2410)*. CEUR-WS.org.

[14] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to information retrieval*. Cambridge University Press.

[15] Saeed Mehrabi, Anand Krishnan, Sunghwan Sohn, Alexandra M. Roch, Heidi Schmidt, Joe Kesterson, Chris Beesley, Paul R. Dexter, C. Max Schmidt, Hongfang Liu, and Mathew J. Palakal. 2015. DEEPEN: A negation detection system for clinical text incorporating dependency relation into NegEx. *J. Biomed. Inf.* (2015).

[16] Roser Morante and Eduardo Blanco. 2012. *SEM 2012 Shared Task: Resolving the Scope and Focus of Negation. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*. Association for Computational Linguistics, Montréal, Canada, 265–274. https://aclanthology.org/S12-1035

[17] Priyanka Nigam, Yiwei Song, Vijai Mohan, Vihan Lakshman, Weitian Allen Ding, Ankit Shingavi, Choon Hui Teo, Hao Gu, and Bing Yin. 2019. Semantic Product Search. In *KDD*. ACM, 2876–2885.

[18] Zhong Qian, Peifeng Li, Qiaoming Zhu, Guodong Zhou, Zhunchen Luo, and Wei Luo. 2016. Speculation and Negation Scope Detection via Convolutional Neural Networks. In *EMNLP*. The Association for Computational Linguistics, 815–825.

[19] Lance A. Ramshaw and Mitch Marcus. 1995. Text Chunking using Transformation-Based Learning. In *VLC@ACL*.

[20] Stephen E. Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.* 3, 4 (2009), 333–389.

[21] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR* abs/1910.01108 (2019).

[22] Daria Sorokina and Erick Cantú-Paz. 2016. Amazon Search: The Joy of Ranking Products. In *SIGIR*. ACM, 459–460.

[23] Andrew Trotman, Jon Degenhardt, and Surya Kallumadi. 2017. The Architecture of eBay Search. In *eCOM@SIGIR (CEUR Workshop Proceedings, Vol. 2311)*.

[24] Quoc-Tuan Truong, Tong Zhao, Changhe Yuan, Jin Li, Jim Chan, Soo-Min Pantel, and Hady W. Lauw. 2022. AmpSum: Adaptive Multiple-Product Summarization towards Improving Recommendation Captions. In *WWW*. ACM, 2978–2988.

[25] Peter D. Turney and Michael L. Littman. 2002. Unsupervised Learning of Semantic Orientation from a Hundred-Billion-Word Corpus. *CoRR* cs.LG/0212012 (2002).

[26] Jun Yan, Nasser Zalmout, Yan Liang, Christan Grant, Xiang Ren, and Xin Luna Dong. 2021. AdaTag: Multi-Attribute Value Extraction from Product Profiles with Adaptive Decoding. In *ACL/IJCNLP (1)*. ACL, 4694–4705.