Designing a Prototype Architecture for Crowdsourcing Language Resources

Christos Rodosthenous
Open University of Cyprus, Nicosia, Cyprus
christos.rodosthenous@ouc.ac.cy

Verena Lyding
Eurac Research, Bolzano/Bozen, Italy
verena.lyding@eurac.edu

Alexander König
Eurac Research, Bolzano/Bozen, Italy
Alexander.Koenig@eurac.edu

Jolita Horbacauskiene
Kaunas University of Technology, Lithuania
jolita.horbacauskiene@ktu.lt

Anisia Katinskaia
University of Helsinki, Finland
anisia.katinskaia@cs.helsinki.fi

Umair ul Hassan
Insight Centre for Data Analytics, National University of Ireland, Galway
umair.ulhassan@insight-centre.org

Nicos Isaak
Open University of Cyprus, Nicosia, Cyprus
nicos.isaak@st.ouc.ac.cy

Federico Sangati
Orientale University, Napoli, Italy
fsangati@unior.it

Lionel Nicolas
Eurac Research, Bolzano/Bozen, Italy
lionel.nicolas@eurac.edu

Abstract

We present an architecture for crowdsourcing language resources from language learners and a prototype implementation of it as a vocabulary trainer. The vocabulary trainer relies on lexical resources from the ConceptNet semantic network to generate exercises while using the learners' answers to improve the resources used for the exercise generation.

2012 ACM Subject Classification Information systems → Web services; Information systems → Ontologies

Keywords and phrases Crowdsourcing, Language Learning, Language Resources, Lexicon, Knowledge Bases, ConceptNet, Commonsense Knowledge, enetCollect

Acknowledgements This paper is based upon work from the European Network for Combining Language Learning with Crowdsourcing Techniques (EnetCollect) COST Action, supported by COST (European Cooperation in Science and Technology)

LDK 2019 - Posters Track.
Editors: Thierry Declerck and John P. McCrae

1 Introduction

We present a prototype architecture for crowdsourcing language resources from language learners and a first implementation of it for creating interactive vocabulary exercises which crowdsource [3] the learners' answers, aiming to improve the language resources used to generate the content of the exercises.

The current architecture is designed to accommodate various language resources, such as mono- and bilingual corpora or lexicons as well as content from commonsense knowledge bases and ontologies. The architecture foresees that exercises can be delivered via several user interfaces thanks to the implementation of a RESTful API approach, allowing the logical separation between computation and presentation layers.

Work presented here is similar to that of *Duolingo*, a platform [14] which is used to crowdsource translations from learners. Other related work includes initiatives using explicit crowdsourcing, which have primarily employed Amazon Mechanical Turk for data collection. For instance (among many others), in [1] the authors created a Turk Bootstrap Word Sense Inventory of frequently used nouns in English.

Also, approaches of implicit crowdsourcing, which mostly rely on Games With A Purpose (GWAPs), relate to the logic underlying the architecture presented here. For example, in [9] a platform that combines automated reasoning with games for acquisition of knowledge rules was developed. Moreover, in [6], a web based game called Common Consensus is described, based on the popular TV game show 'Family Feud'. That game is used to collect and validate commonsense knowledge about everyday goals.

The proposed architecture as well as a vocabulary trainer prototype and its features are presented in the following sections. The code of the project resides on GitLab[1] for interested readers to test it or even more, help in expanding it.

2 Implicit Crowdsourcing Paradigm

The presented prototypical architecture builds on an implicit crowdsourcing paradigm which follows the idea that:

`IF` a language resource can be used to automatically generate language learning exercises,
`THEN` learners' answers to these exercises can also be used to improve the resource.

This paradigm thus exploits a win-win strategy [7]) between people in need of high quality language resources and people in need of online language learning material. It bootstraps a virtuous circle between both parties, where the answers of the learners allow the enhancement of the language resources, which in turn will result in higher quality learning content, due to the fact that it is generated from the improved language resources.

Such a paradigm can be applied to any scenario in which a language resource (e.g., treebanks, wordnets, corpora) can be paired with a specific language learning exercise, in the sense that the exercise content can be automatically generated from the LR.

There is a somewhat counter-intuitive aspect to this paradigm: the assumption that a crowd of learners, with their natural deficiencies regarding their knowledge of the language, can be of use for improving language resources–a task that is usually performed by expert

[1] `https://gitlab.com/crowdfest_task3`

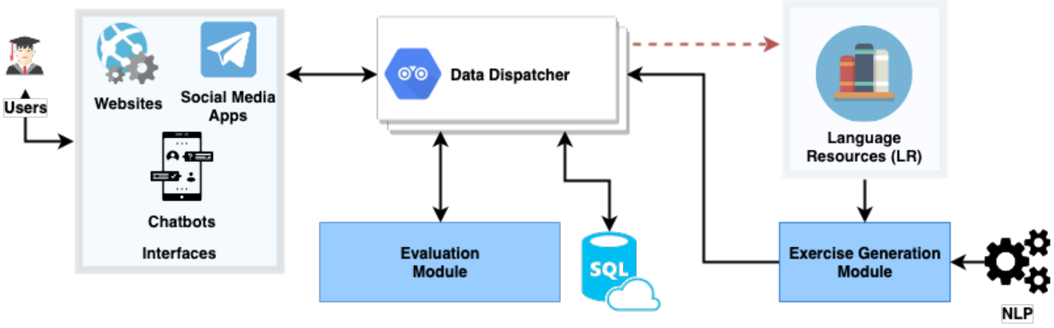


Figure 1 An overview of the proposed architecture, presenting the core modules of the platform, the data interchange between them and user interactions. The red dashed-line arrow represents the update of the initial Language Resource with crowd-contributed data.

linguists.[2]

However, the lack of expertise of the crowd can be compensated in two ways: (1) by continuously evaluating the performance of the learners and taking it into account, and (2) by cross-matching judgments from their answers to the exercise questions.

Regarding (1), evaluating the learner is considered feasible, as in most cases the learning application should not crowdsource on the learner, but provide exercise content that is of satisfying quality and should thus be generated from reliable LR entries considered as 'gold standard'. Accordingly, learners can be evaluated on this gold standard content, while we crowdsource their answers only on new or unreliable entries, and at a very moderate rate (e.g., applying a ratio of 95% of reliable exercise content vs. 5% of exercises to crowdsource new content).

Regarding (2), cross-matching judgments of learners to deduce the correct answers can be addressed by an aggregation approach which relies on both the classic trade-off between quantity and quality (a low quality of answers is made up for by a higher quantity of answers), and the possibility to decompose any complex question in smaller grained elements that can be asked to learners through a set of boolean questions (e.g., 'Does the learner believe that the French word "manger" is a verb?'). Indeed, provided that the crowdsourced answers allow to directly or indirectly deduce a boolean opinion, then all answers from learners with performances superior to 50% to such a task allow to progress towards statistical certainty. Therefore, one only needs to keep on asking the same question to different learners until the set of answers allows to reach a statistical threshold ensuring good quality (e.g., a reliability score above 98%).

3 System Architecture

The proposed architecture is based on a modular schema and comprises four modules: (1) an exercise generation module, (2) a data dispatcher, (3) an evaluation module and (4) one or several user interfaces. In Figure 1, a high-level overview of the architecture is depicted showing the core modules and processes.

The **exercise generation module** is responsible for content retrieval from any type of language resource (LR) like corpora, knowledge bases and lexica, which contain language data

[2] Readers can picture it as asking a group of tourists for a route in the city they are visiting.

in a structured form. It handles the retrieval of specific data from a LR, e.g., all collocates of the word ‘challenge’ from a collocation lexicon, and automatically processes them in order to create exercises. The processing could include grouping the collocates by word class, normalize singular and plural forms of substantives, etc. The exercise generation module delivers the exercises to the data dispatcher which can deliver valid answers to the exercises, back into the LR. Furthermore, it can also use natural language processing techniques to convert data, e.g., extracting the lemma of a word.

The **data dispatcher module** handles all transactions between the various modules. It caters for receiving and passing on data in a generic exchange format (such as JSON[3]). For example it may receive generated exercises of different types and passes them on to multiple user interfaces. In return it receives back the response from the completed exercises from the user interfaces and passes them on to the evaluation module. After receiving the processed results, it can return the crowdsourced data to the original language resource that the exercises were generated from. The whole communication is done through secure web-service transactions between the various modules.

The **evaluation module** processes the results retrieved from learners when completing exercises. Different types of aggregation methods can be applied to determine correct and wrong answers. This validation information is used for two purposes in the presented architecture: firstly to update or enhance the LR with new generated (crowdsourced) information and secondly to provide feedback to the learners about their performance while completing the exercises.

The **user interaction module** can handle integrations of the data dispatcher with different user interfaces such as chatbots (e.g., Telegram[4]) and web-based applications. The architecture can be utilized by any user interface that is able to consume the exercises structure, data and incentive mechanism through its API, while preserving the same logic behind the exercise.

4 Vocabulary Trainer

As a first implementation of the prototype architecture for crowdsourcing language resources, we present an interactive vocabulary trainer, which is built using data from ConceptNet, a commonsense semantic network [13]. It offers vocabulary exercises to practice semantic relations between words.

In language learning, vocabulary enhancing exercises based on words semantic relations are considered to be effective activities. In [11], the aspects of background knowledge, context and morphology to learn words more effective and clarify word meaning as essential to vocabulary instruction are presented. The richness of acquired vocabulary depends not only on the number of learned lexical items but also from the ability to connect and share semantic networks of similar concepts. Authors of [2] argue that *‘word learning is not simply the process by which isolated object– label associations are added to the mental lexicon one by one but also involves the learning of interrelated clusters of concepts, in which the knowledge of one concept supports the learning of another’* (p. 42).

ConceptNet is a large semantic network that describes general human knowledge and how it is expressed in natural language. Facts in ConceptNet originate from sources like DBPedia

[3] `https://www.json.org/`

[4] `https://core.telegram.org/bots/api`

[5], Wiktionary[5] and popular GWAPS and crowdsourcing projects, such as Verbosity [15], the Open Mind Common Sense project [12], etc.

The **exercise generation module** is responsible for retrieving content from ConceptNet and for creating language learning exercises from the retrieved content. This is done by quering directly the conceptnet.io APIs for relevant content.

ConceptNet provides a large set of background knowledge about different facts connected with other facts using relations such as `relatedTo`, `AtLocation`, `PartOf`, `IsA`, etc.

For instance, if a search for knowledge that `relatesTo` the term 'cat' is initiated, ConceptNet will return results such as 'feline', 'pet', 'dog', etc. Afterwards, the exercise generation module processes the results using a natural language processing application to remove stopwords and duplicates, retrieve lemmas and store them in a database.

An example of a generated exercise is 'Name one thing that is related to cat', where the learner is expected to provide a word that exists in the results retrieved from the knowledge base. In cases where new words are added, the evaluation module checks whether they should be added to the knowledge base or not, while a specific user feedback strategy is used to account for the unknown correctness of the answer.

The **data dispatcher module** of the vocabulary trainer is handling transactions between the various modules by using secure web-services, where requests are received and the outcome is presented in JSON[6] format. Detailed specification of the API is available at the project repository. The architecture provides web-services for: (1) registering new users, (2) retrieving exercises from the exercise generation module, (3) checking learners' contributions, (4) assigning points and awards to learners, and (5) updating the leaderboard.

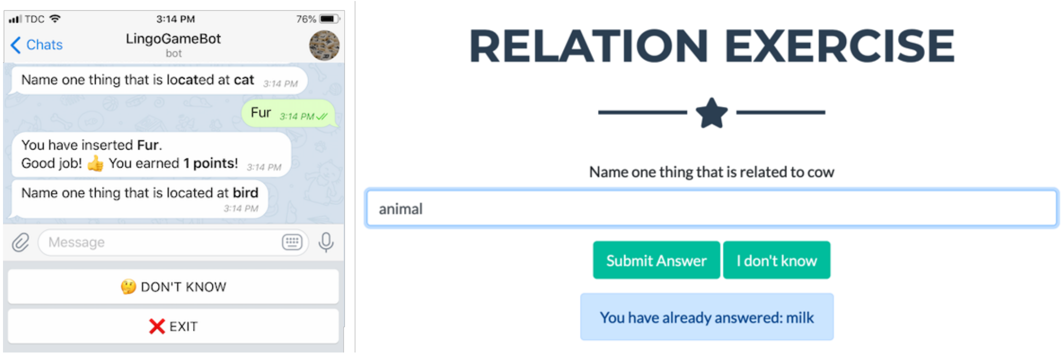
Within the vocabulary trainer, the **evaluation module** processes the learners' answers in order to both update the knowledge base with new words and to assign points and badges to the learners and make the whole process interesting.

Whenever a learner completes an exercise, the evaluation module validates the provided answer against the knowledge base. If the answer is already part of the knowledge base then the learner receives points. If the answer is not part of the knowledge base then it is put on hold until a certain number of *new* words (i.e., candidate words for the knowledge base) have been accumulated for that specific exercise. In the second case, the learner receives a feedback message explaining that there are additional points *pending* to be approved.

Once the pre-defined threshold of *new* words is met the list of candidate words is ranked according to the frequency and feedback is sent to each learner who provided an answer. For the highest-ranked word among the list of candidates the *pending* points are turned into actual points and the knowledge base is updated with the word. All learners who had provided that answer are awarded points, and the learners who provided that answer first, receive also an additional badge.

The **user interaction module** is currently populated by two user interfaces in the vocabulary trainer: (1) The Telegram messenger chatbot, and (2) a web application using the popular Bootstrap framework (see Figure 2). Both interfaces use APIs to communicate with the data dispatcher, query it for new exercises, display these to the user and store the user's answer. The generic architecture ensures that both interfaces can implement the same features while presenting them to the user in different ways.

[5] `https://www.wiktionary.org/`
[6] `https://www.json.org/`



Name one thing that is located at **cat** 3:14 PM

Fur 3:14 PM

You have inserted **Fur**.
Good job! 👍 You earned **1 points!** 3:14 PM

Name one thing that is located at **bird** 3:14 PM

RELATION EXERCISE

Name one thing that is related to cow

animal

You have already answered: milk

Figure 2 A screenshot depicting the two prototype implementations, i.e., the Telegram chatbot (left) and the Bootstrap web application (right) where a user is contributing new words, while completing exercises.

5 Conclusion and Future Work

In this paper, we presented an architecture to crowdsource language resources from language learning exercises delivered via several user interfaces. The proposed architecture is versatile and expandable and it is not restricted to a specific paradigm or dataset. Different language resources can be used for generating learning content and several types of exercises can be added. Also different evaluation strategies to cross-match learners' answers can be incorporated to accept or reject an answer and update the corresponding language resource.

Furthermore, we presented the first prototype implementation on top of the architecture, i.e., a vocabulary trainer that relies on ConceptNet to deliver exercises. Early tests with both the Telegram chatbot and the Bootstrap web application show promising results in terms of acquisition of knowledge facts and usefulness of the architecture for that purpose.

We are currently designing an experiment to formally evaluate all components of the architecture. We also plan to deliver the exercises via the language learning platform Revita [4] and existing knowledge-based GWAPs [8, 9]. Future directions of our research could also include exercises related to geography, which can be used to populate a knowledge base for identifying the geographic focus of a text [10], using words that are related to a specific geographic location, e.g., feta `RelatedTo` greece.

References

1 Chris Biemann. Creating a system for lexical substitutions from scratch using crowdsourcing. *Language Resources and Evaluation*, 47(1):97–122, 2013.

2 Elizabeth Hadley, David Dickinson, Kathy Hirsh-Pasek, and Roberta Golinkoff. Building semantic networks: The impact of a vocabulary intervention on preschoolers' depth of word knowledge. *Reading Research Quarterly*, 54:41–61, 01 2018. `doi:10.1002/rrq.225`.

3 Jeff Howe. Crowdsourcing: A Definition, 2006. URL: `http://www.crowdsourcing.com/cs/2006/06/crowdsourcing{_}a.html`.

4 Anisia Katinskaia, Javad Nouri, and Roman Yangarber. Revita: a language-learning platform at the intersection of its and call. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, 2018.

5 Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195, 2015.

6 Henry Lieberman, Dustin A Smith, and Alea Teeters. Common Consensus: A Web-Based Game for Collecting Commonsense Goals. In *Proceedings of the Workshop on Common Sense and Intelligent User Interfaces*, Honolulu, Hawaii, USA, 2007.

7 Lionel Nicolas, Verena Lyding, Luisa Bentivogli, Federico Sangati, Johanna Monti, Irene Russo, Roberto Gretter, and Daniele Falavigna. Enetcollect in italy. In *CLiC-it*, 2018.

8 Christos Rodosthenous and Loizos Michael. Gathering background knowledge for story understanding through crowdsourcing. In *Proceedings of the 5th Workshop on Computational Models of Narrative (CMN 2014)*, volume 41, pages 154–163, Quebec, Canada, 2014. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. `doi:10.4230/OASIcs.CMN.2014.154`.

9 Christos Rodosthenous and Loizos Michael. A hybrid approach to commonsense knowledge acquisition. In *Proceedings of the 8th European Starting AI Researcher Symposium*, pages 111–122, 2016. `doi:10.3233/978-1-61499-682-8-111`.

10 Christos Rodosthenous and Loizos Michael. Using generic ontologies to infer the geographic focus of text. In Jaap van den Herik and Ana Paula Rocha, editors, *Agents and Artificial Intelligence*, pages 223–246, Cham, 2019. Springer International Publishing.

11 Catherine Rosenbaum. A word map for middle school: A tool for effective vocabulary instruction. *Journal of Adolescent Adult Literacy*, 45(1):44–49, 01 2001.

12 Push Singh, Thomas Lin, Erik T. Mueller, Grace Lim, Travell Perkins, and Wan Li Zhu. Open mind common sense: Knowledge acquisition from the general public. In Robert Meersman and Zahir Tari, editors, *On the Move to Meaningful Internet Systems 2002: CoopIS, DOA, and ODBASE*, pages 1223–1237, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

13 Robert Speer and Catherine Havasi. Representing general relational knowledge in conceptnet 5. In *LREC*, pages 3679–3686, 2012.

14 Luis Von Ahn. Duolingo: learn a language for free while helping to translate the web. In *Proceedings of the 2013 international conference on Intelligent user interfaces*, pages 1–2. ACM, 2013.

15 Luis von Ahn, Mihir Kedia, and Manuel Blum. Verbosity: A Game for Collecting Common-Sense Facts. In *Proceedings of the 25th SIGCHI Conference on Human Factors in Computing Systems (CHI 2006)*, page 75, Montréal, Québec, 2006. ACM. `doi:10.1145/1124772.1124784`.