Post-print of: Boiński T., Szymański J. (2020) Collaborative Data Acquisition and Learning Support. In: Saeed K., Dvorský J. (eds) Computer Information Systems and Industrial Management. CISIM 2020. Lecture Notes in Computer Science, vol 12133. Springer, Cham. DOI: https://doi.org/10.1007/978-3-030-47679-3_19

Collaborative data acquisition and learning support

Tomasz Boiński[0000−0001−5928−5782] and Julian Szymański[0000−0001−5029−6768]

Faculty of ETI, Gdańsk University of Technology, Gdańsk, Poland
{tomasz.boinski, julian.szymanski}@eti.pg.edu.pl

Abstract. With the constant development of neural networks, traditional algorithms relying on data structures lose their significance as more and more solutions are using AI rather than traditional algorithms. This in turn requires a lot of correctly annotated and informative data samples. In this paper, we propose a crowdsourcing based approach for data acquisition and tagging with support for Active Learning where the system acts as an oracle and repository of training samples. The paper presents the CenHive system implementing the proposed approach. Three different usage scenarios are presented that were used to verify the proposed approach.

Keywords: Data annotation · Annotation verification · Active Learning.

1 Introduction

With the constant development of neural networks, traditional algorithms relying on data structures lose their significance. More and more problems, even those solved using traditional algorithms, due to the sheer size of the problem space, rely on deep neural networks. Teaching neural networks in turn rely heavily on data acquisition, tagging and the ability to increase the network quality rather than pure algorithms creation and optimization.

The difficulty of data acquisition itself varies depending on the domain and usually cannot be easily automated. Furthermore without annotation the data is usually useless. The process of data annotation or tagging can be time-consuming and requires human effort. For example we can quite easily record bees entering and exiting a hive. Tagging the images correctly marking all the bees in the pictures is however a very difficult task. This can be done correctly either by a very refined neural network (which had to be prepared beforehand and required an annotated set of training for the learning process) or a team of humans. In some cases the humans involved have to be experts in the field.

In this paper, we propose a crowdsourcing based approach for data acquisition and tagging with support for Active Learning [11, 8] where the system acts as an oracle and repository of training samples. The training samples, as they can be the neural network output verified by humans, can serve as correctional samples that were previously wrongly classified by the network.

The structure of this paper is as follows. In Section 2 available approaches to data acquisition are presented. Next, in Section 3 architecture of the proposed system for data acquisition and verification is described. Later on in Section 4 usage of the proposed system is presented in terms of data acquisition and verification. Section 5 discusses system usage for Active Learning support and finally a summary is given.

2 Data acquisition and annotation approaches

Crowdsourcing-based data acquisition and annotation are well established. Companies, like VoiceLab[1], often rely on crowdsourcing for gathering real-life data (e.g. voice recordings of an average human rather than a professional speaker). For such solutions, dedicated systems are created. Another example of such approach is Snapshot Serengeti[2] project [12], where users can browse pictures of the park's wildlife captured by automated cameras and describe the contents of the images in terms of the number and type of animals visible. The project was created to help track the migration habits of the park animals. This solution is based purely on the will of the community to help in solving the problem, the only reward is the possibility of watching the wildlife. Other solutions like Duolingo[3] [5] aim at providing commercial data transformation. In this case the users, while learning the foreign language of their choice, perform translation of texts that are further used by news agencies.

One of the first and the most general examples of crowdsourcing data acquisition is, however, Amazon Mechanical Turk[4] [3]. This platform allows the creation of tasks that can vary from simple data annotation (e.g. tags creation or verification) to data creation (e.g. review writing, providing voice recordings etc.). The platform allows giving the users some small financial gratitude for every task solved. The money is paid by the problem supplier. Other solutions, like Google reCaptcha [16] or Foldit [2] are proof of the viability of this approach, especially that all aforementioned systems proved to be successful.

Currently, available solutions are usually very specific. They are used to solve a one, well-defined problem, provide one, dedicated interface or are commercial solutions, usually designed for certain companies.

During our works we designed and implemented a universal service called CenHive [5], that aims at providing a general solution allowing acquisition and annotation of different data types, like any combination of text, images, audio and video. Furthermore the data can be easily distributed to the users for verification through multiple clients. The solution allows also usage of the data acquired for active learning through simple REST interfaces.

[1] https://www.voicelab.ai/

[2] https://www.snapshotserengeti.org/

[3] https://en.duolingo.com/

[4] https://www.mturk.com/mturk/welcome

[5] https://kask.eti.pg.gda.pl/cenhive/

3 CenHive architecture

CenHive is a web-based solution. Originally it was designed as a service dedicated to Wikipedia-WordNet mappings verification [1, 13, 7] using crowdsourcing approach via Games with a Purpose [15]. It supported only one client and only text questions with a fixed amount of answers.

Currently the system has been extended considerably. The current version allows multiple clients and data types. It supports not only text data but audio, video, images and any combination of aforementioned input types both in questions and answers. The amount of answers sent to the client is also configurable and the client, instead of an answer to a question presented, can create a new set of data for verification.

The main part of the system is a database containing the data aggregated into so-called *contents*. It can be a form of mappings (e.g. Wikipedia – WordNet mappings), tags (e.g. pictures with tags describing the content or images with yes/no tags answering *contents* questions, like ”Is this a human face?” or ”Is this a bee?”) or plain data that can be used for detailed data acquisition (like crowd or beehive pictures, camera footage, etc.). The tags or descriptions associated with given *content* are called *phrase*.

The *contents* elements are distributed to CenHive clients with client-defined number of randomly selected *phrases* associated with given *content* using RESTful [10] API calls.

The system supports multiple clients that can use any technology. Currently, the system has 4 official clients:

- TGame[6] – Android platform game originally designed for Wikipedia – WordNet mappings verification, currently supporting audio, video, text and images as elements within the *contents* and *phrases* parts. The player has to answer provided questions to activate checkpoints in the game;
- Truth or Bunk[7] – web-based client designed for Wikipedia – WordNet mappings – it takes a form of a quiz game where players have to answer as many questions as possible;
- 2048 clone[8] – an extension of the original 2048 game[9]. The player, during the game, donates some of his or her computer computation power to create new data (in this case finding human faces on photos provided from CenHive) in volunteer computing mode and can also verify the detected faces using crowdsourcing model when he or she wants to undo a move;
- captcha element – used for securing user login and registration page, can provide a set of questions based on *content* elements with tags stored as *phrases* in the system.

There are 2 other clients in the works that will support general image tagging and will help to produce data for teaching neural networks using an active

[6] https://play.google.com/store/apps/details?id=pl.gda.eti.kask.tgame
[7] https://kask.eti.pg.gda.pl/cenhive/tob/
[8] https://kask.eti.pg.gda.pl/cenhive/2048/
[9] http://git.io/2048

learning approach. Both aim at allowing users verification of bee detection done by a neural network and the ability to tag bees on random images.

4 Data acquisition and verification

The main usage of the system is data verification. CenHive was created as crowdsourcing based solution for Wikipedia – WordNet mappings verification and was later extended for general relations verification. In this case a mapping can be treated as a relation, where a Wikipedia article is a description for a WordNet synset. Relations can thus take any form – keyword-based tags, mappings, yes/no answers, image to text mappings, etc. Currently we are aiming at automatic usage of gathered data for deep neural network learning. In all cases the data can be in any form supported by CenHive.

4.1 Wikipedia-WordNet mappings verification

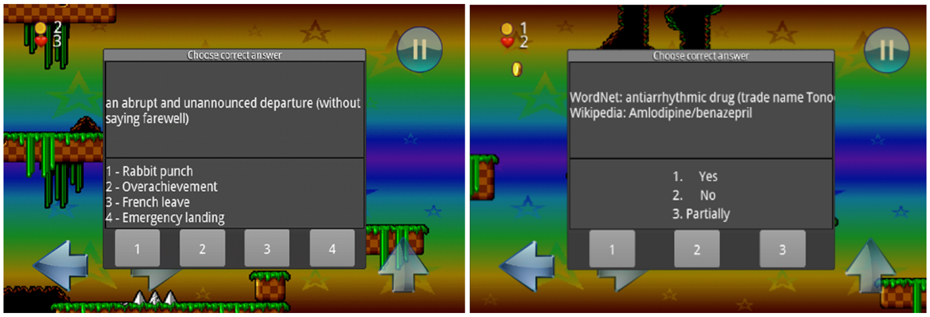
The system was used for assessing and verification of mappings between WordNet synsets and Wikipedia articles. The task was done using a 2D platform game called TGame [1, 13] ("Tagger Game"). It followed the output-agreement model [15]. We aimed at introducing higher replayability by providing a trophy system based on the player in-game performance; both in the form of collectibles gathering, like coins or hearts, and in the number of tasks solved.

In TGame activating a checkpoint requires reaching it and answering the question provided by CenHive, which in turn is a mapping verification. The checkpoint is activated when the user answer (the mapping) is similar to the one in CenHive. All the answers are logged so if the players will mark other answers the administrator can correct the answer. The players also have a chance of reporting the questions thus strongly indicating wrong mapping. We asked for mappings using two approaches:

- we extended the automatically generated mappings with other, false mappings based on the Wikipedia search functionality, the users should choose the correct mapping;
- we presented only the mapping from the database, the user should choose whether it is correct or not.

During two months-long test period, players gave 3731 answers in total to 338 questions. The total number of answers for different mapping types is shown in Table 1.

The tests proved that the proposed solution is useful in mappings verification. During the evaluation, however, we had to deal with malicious users. As in the crowdsourcing we cannot verify the user we decided that only questions that had multiple answers will be taken into consideration. Furthermore to eliminate blind shots we decided to take into account only questions that the user had displayed for at least 5 seconds. Such time proved to be enough for the user to actually read the question and the answers presented. Fortunately the number



an abrupt and unannounced departure (without saying farewell)

1 - Rabbit punch
2 - Overachievement
3 - French leave
4 - Emergency landing

WordNet: antiarrhythmic drug (trade name Tono
Wikipedia: Amlodipine/benazepril

1. Yes
2. No
3. Partially

Fig. 1. TGame user interface (left: extended mappings, right: yes/no question).

Table 1. Number of answers for different type of questions

Question type	Questions	Answers	Answers per question	Reports	Blind shots	Blind shots %
Extended mappings	239	3,308	13.84	625	16	0.48
Yes/No	99	423	4.27	10	12	2.84
Total	338	3731	11.04	685	62	1.34

of answers that could be considered blind shots is very low, in worst case it did not go above 3% and across all tests the ratio of blind shots was no greater than 1.56%.

4.2 Face recognition and verification

The first major extension introduced into CenHive was the ability to not only verify already gathered data but to create the data itself. This can be done twofold – the users can either generate data manually, e.g. by mapping the images with a dedicated client, or provide the computation power in the volunteer computing model. During this stage we decided to use the second approach. For that we created a dedicated client in the form of a 2048 game clone. It is available as a web application, where, during the gameplay, the photos are downloaded from the CenHive server. The photos are then analyzed using Viola-Jones Feature Detection Algorithm using Haar Cascades [14] algorithm (more specifically using HAAR.js implementation[10]). Detected faces are then sent back to CenHive and stored as tags where the original image is tagged with coordinates of the rectangle containing the detected face (Fig. 2).

During the test period, where tests were done with the help of the research team and students, for 64 multi-person photos the game players generated 628 face-detects (giving 303 distinct face detects). In this case multiple face detects

[10] https://github.com/foo123/HAAR.js

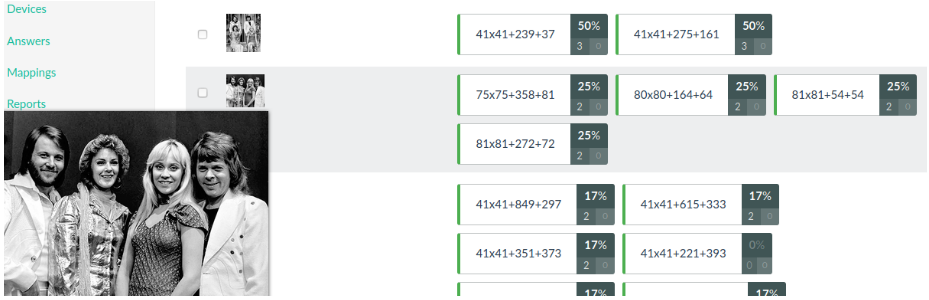


Fig. 2. Detected faces.

are not necessary as they are done arithmetically using the player's device processing power. The detected faces were further verified – for 92 images we got 181 verification responses and 7 reports.

5 Active learning support

Traditionally when performing a deep network learning for detection problems a big learning set is needed. The data for teaching is randomly selected, which can lead to longer teaching times as newly selected samples can contain partial or even no new information. This leads to a small or even lack of increase in network quality.

This problem can be mitigated with the Active Learning approach [8, 11, 4]. In this approach the network should choose whether the sample is informative enough for the sample to be a part of the teaching set.

In our approach CenHive is treated as an oracle that should be able to answer the question of whether the given sample is informative or not. In our test case (bee detection on video streams) we consider a sample informative when the network wrongly detects bees on the given image.

For each step of Active Learning we select samples based on the following criteria:

- For each sample, detection is done using the network from the previous iteration, only tags with the detection certainty over given value were taken into account,
- Tagging done by the network is compared to the ones from the oracle
  - If the number of objects detected by the network differs from the number of objects returned by the oracle the sample is added to the training set,
  - If the number of detected objects is the same, for each object we calculate Intersection over Union (IoU) value, meaning how well the area of detected objects covers the area of objects returned by the oracle. If it is below the predefined threshold we add the sample to the training set.

In this case the oracle works as described in the previous chapter. The CenHive system distributes neural network tagged images to the users for verification. The implemented clients also allow manual image tagging allowing thus the creation of new training data. Using friends and family approach we managed to tag 2500 photos, what after splitting the images to the size appropriate for the network gave 60423 samples.

To test the viability of the proposed solution we performed Active Learning on Faster R-CNN [9] network implemented using Detectron[11] [6] system. The network was first trained using 12000 random pictures during 13 epochs. During the tests we considered two parameters: IoU value over 70% and over 80% and minimal certainty score over 30% and over 60%. Three models were thus trained:

- min IoU 70%, min score 30% – the image was considered correctly detected by the network when IoU had value at least 70% and certainty score was over 30%,
- min IoU 80%, min score 30% – the image was considered correctly detected by the network when IoU had value at least 80% and certainty score was over 30%,
- min IoU 80%, min score 60% – the image was considered correctly detected by the network when IoU had value at least 80% and certainty score was over 60%.

The images tagged by the network were then verified using CenHive and the network was trained again for 12 iterations with the extended training set. For each iteration up to 1000 new images verified using CenHive were added to the training set. During each iteration The aforementioned approach showed constant network quality upgrade, however with each new data set from CenHive the training set grew and thus the training time increased. We then tested the approach with limited usage of the old training set. Altogether we used three approaches to introduce new samples:

- Full training set – the training set for given iteration consisted of new samples and all previously used samples,
- Only new data – the training set for given iteration consisted of only the new images verified through CenHive,
- New data with memory – the training set contained new images and 10000 randomly selected images from previous learning sessions.

The results are presented in Fig. 3. Using only new samples did not provide good results. However when the size of the training set was limited to new samples and 10000 randomly selected previously used the network quality was nearly as good as when the training was done with all the samples. The training time, however, was much smaller, as can be seen in Fig. 4. The data acquired proves that the proposed approach is viable to serve as an oracle in the Active Learning approach.

[11] https://github.com/facebookresearch/Detectron

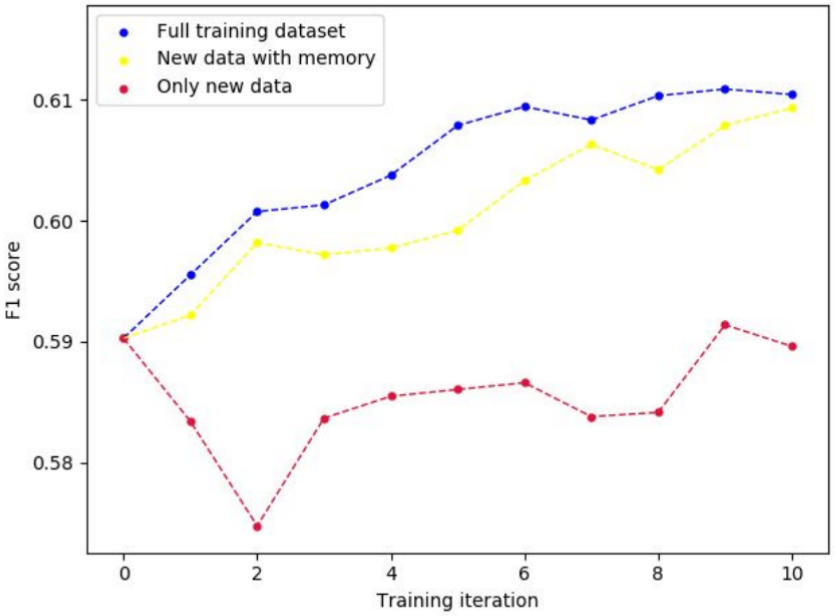


Fig. 3. Network quality.

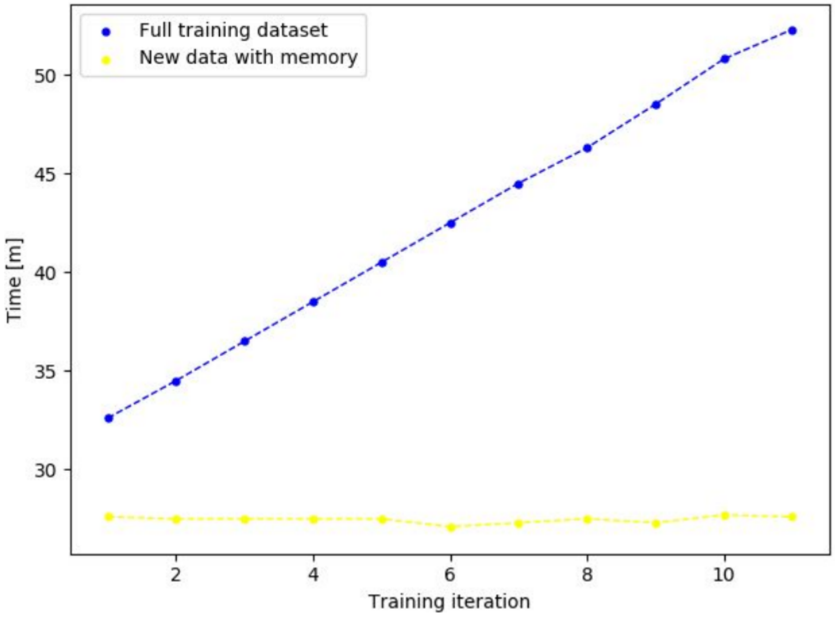


Fig. 4. Training times.

6 Summary and future work

We believe that with each day we are more justified to state, that modern solutions are less algorithms and more annotated data. More and more modern problems cannot be easily solved using traditional approaches and require the usage of deep neural networks which in turn require a high amount of data for a precise learning process.

The paper presents a crowdsourcing based approach to data gathering, annotation and verification. The approach also allows automatic usage of gathered data for Active Learning where the implemented system acts as an oracle.

We tested the approach for different data types and tasks. The approach proved to be viable in data gathering, annotation and verification. We also showed that it can be used effectively in the Active Learning as an oracle that determines whether the given sample is informative and should be used in the learning process or not.

The drawback of the solution is that it relies on heavy community involvement. This can be however achieved by embedding the problems in so-called Games with a Purpose. We implemented a few such clients and performed preliminary tests proving such approach viable.

In the future we plan on extending the solution with more and better-suited clients. We also plan on using captcha element in real-life systems that will guarantee a steady stream of solutions provided by the users of such systems. This will allow us optimization of the whole verification process generating more training data for deep learning.

Acknowledgements

We would like to thank Agata Krauzewicz and Łukasz Łepek who implemented part of the presented solution during their studies.

References

1. Boiński, T.: Game with a purpose for mappings verification. In: Computer Science and Information Systems (FedCSIS), 2016 Federated Conference on. pp. 405–409. IEEE (2016)
2. Curtis, V.: Motivation to participate in an online citizen science game: A study of foldit. Science Communication **37**(6), 723–746 (2015)
3. Fort, K., Adda, G., Cohen, K.B.: Amazon mechanical turk: Gold mine or coal mine? Computational Linguistics **37**(2), 413–420 (2011)
4. Gal, Y., Islam, R., Ghahramani, Z.: Deep bayesian active learning with image data. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 1183–1192. JMLR. org (2017)
5. Garcia, I.: Learning a language for free while translating the web. does duolingo work? International Journal of English Linguistics **3**(1), 19 (2013)
6. Girshick, R., Radosavovic, I., Gkioxari, G., Dollár, P., He, K.: Detectron. https://github.com/facebookresearch/detectron (2018)

7. Jagoda, J., Boiński, T.: Assessing word difficulty for quiz-like game. Semantic Keyword-Based Search on Structured Data Sources: IKC: International KEYSTONE Conference on Semantic Keyword-Based Search on Structured Data Sources **10546**, 70–79 (Jan 2018). https://doi.org/10.1007/978-3-319-74497-1_7
8. Kellenberger, B., Marcos, D., Lobry, S., Tuia, D.: Half a percent of labels is enough: Efficient animal detection in uav imagery using deep cnns and active learning. IEEE Transactions on Geoscience and Remote Sensing **57**(12), 9524–9533 (2019)
9. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. pp. 91–99 (2015)
10. Richardson, L., Ruby, S.: RESTful web services. " O'Reilly Media, Inc." (2008)
11. Settles, B.: Active learning literature survey. Tech. rep., University of Wisconsin-Madison Department of Computer Sciences (2009)
12. Swanson, A., Kosmala, M., Lintott, C., Simpson, R., Smith, A., Packer, C.: Snapshot serengeti, high-frequency annotated camera trap images of 40 mammalian species in an african savanna. Scientific data **2**, 150026 (2015)
13. Szymański, J., Boiński, T.: Crowdsourcing-based evaluation of automatic references between wordnet and wikipedia. INTERNATIONAL JOURNAL OF SOFTWARE ENGINEERING AND KNOWLEDGE ENGINEERING **29**(03), 317–344 (Jan 2019). https://doi.org/10.1142/s0218194019500141
14. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001. vol. 1, pp. I–I. IEEE (2001)
15. Von Ahn, L.: Games with a purpose. Computer **39**(6), 92–94 (2006)
16. Von Ahn, L., Maurer, B., McMillen, C., Abraham, D., Blum, M.: recaptcha: Human-based character recognition via web security measures. Science **321**(5895), 1465–1468 (2008)