Conversing Learning: active learning and active social interaction for human supervision in never-ending learning systems

Saulo D. S. Pedro and Estevam R. Hruschka Jr.

Federal University of Sao Carlos, UFSCar Sao Carlos, Brazil

Abstract. The Machine Learning community have been introduced to NELL (Never-Ending Language Learning), a system able to learn from web and to use its knowledge to keep learning infinitely. The idea of continuously learning from the web brings concerns about reliability and accuracy, mainly when the learning process uses its own knowledge to improve its learning capabilities. Considering that the knowledge base keeps growing forever, such a system requires self-supervision as well as self-reflection. The increased use of the Internet, that allowed NELL creation, also brought a new source of information on-line. The social media becomes more popular everyday and the AI community can now develop research to take advantage of these information, aiming to turn it into knowledge. This work is following this lead and proposes a new machine learning approach, called *Conversing Learning*, to use collective knowledge from web community users to provide self-supervision and self-reflection to intelligent machines, thus, they can improve their learning task. The Conversing Learning approach explores concepts from Active Learning and Question Answering to achieve the goal of showing what can be done towards autonomous Human Computer Interaction to *automatically* improve machine learning tasks.

1 Introduction

Machine Learning (ML) has been an effervescent research topic in the last years. New algorithms and new approaches have been proposed bringing relevant contribution to the AI community in general, and also, enhancing learning capabilities of computational systems. A new and relevant approach for machine learning systems is called *Active Learning* (AL). The basic principle behind active learning [1] [2] is to improve machine learning algorithms performance by selecting specific training data. Following along these lines, active machine learning algorithms can achieve better accuracy with fewer training instances if they can choose the data from which they learn. In this sense, an active learning system should identify the most relevant training instances, and then, use them in its learning process.

When exploring principles behind AL, some researchers have proposed the idea of ***Inter****active Learning* (IL) [3] [4] where AL is performed not only once, and the learning process is continuous during a number of iterations. In such an approach, after each iteration, the system interacts with the user and may pose queries (usually in the

[0] the authors would like to thank FAPESP for supportng this work.

form of unlabeled data instances to be labeled) that will help improving the learning results in the next iteration.

Another recent and relevant research topic in Machine Learning is the *Never-Ending Learning* approach that focus on proposing algorithms and models to build learning systems that learn cumulatively forever, using what they have learned yesterday to improve their ability to learn better today, and keep learning indefinitely. The first Never-Ending Learning system described in the literature was proposed in [5] and is called NELL (Never-Ending Language Learner). NELL has been continuously running since January 2010, attempting to perform two main tasks each day [1]: first, it attempts to *read* or extract facts from text found in hundreds of millions of web pages (e.g., playsInstrument(George_Harrison, guitar)). Second, it attempts to improve its reading competence, so that tomorrow it can extract more facts from the web and more accurately than today. So far, NELL has accumulated over 15 million candidate beliefs by reading the web, and it is considering these at different levels of confidence.

In addition to the aforementioned ideas and approaches, the quick development of new technologies in communication and in data storage and processing allows companies to deliver better quality and widespread projects in sharing content and communication. The popularity and power of social media connects more people everyday and all kinds of subjects are discussed worldwide through social media applications. All these factors together resulted in a growing interest of Artificial Intelligence (AI) and ML researchers in exploring web communities to solve new and traditional problems. As already presented by [6], the information available in social web has potential to be turned into high valued content.

We believe that the conjunction of all previously discussed achievements put us in a privileged position to propose an new type of learning system, that takes advantage of Active Learning, Interactive Learning, Never-Ending Learning techniques and the Web Communities. In this sense, we believe that Never-Ending Learning systems can go beyond IL and can autonomously search (in a proactive way) for human supervision on the Web whenever it needs to confirm any information (or to label training instances). Therefore, even if an user cannot give any feedback to the system (as users do in an ***Inter****active Learning* approach), the system should be capable of autonomously finding answers from other sources (i.e. on the web communities), thus, helping to refine and improve its learning capabilities even when a specific user cannot give any feedback. In this work, we call such a system: Conversing Learning System. In such a system, the acquired knowledge can be exploited to help supervision and knowledge revision tasks. Thus, the proposed approach allows self-supervision and self-reflection for learning systems. The possibility of self-supervision, self-reflection and knowledge revision is even more relevant in systems that learn forever like NELL. It is interesting to mention, that looking for answers or feedback on specific themes on web communities is a natural behavior of many human Internet users. Many people ask questions on specific forums (on the Web), as well as, many other ones offer advice and guidance. In this work, we intend to show how a Never-Ending Learning system (like NELL) can *autonomously*

[1] http://rtw.ml.cmu.edu

use the content available on two Web communities (Yahoo!Answers[2] and Twitter[3]) to bring better quality and accuracy to its learning methods.

The main contributions of this paper are: (i) Presenting a *Conversing Learning* system that can be used to help NELL to be self-supervised and to autonomously review its knowledge base (KB) contents; and (ii) Exploring the use of two different Web communities and discuss the main differences regarding the conversing learning aspects.

2 Related Work

NELL is the first Never-Ending Learning system described in the literature. It was developed at Carnegie Mellon University [5] and uses its acquired knowledge to learn better each day. The research team fed the system with an initial ontology and initial seeds. The system, then, takes advantage of the combination of several algorithms to continuously induce new knowledge from millions of web pages. The combination of the algorithms is itself a kind of self-supervision, but the system also counts on some shallow human supervision to ensure it is free from errors, thus avoiding concept drifting.

Even not focusing on the definition of a Conversing Learning system, the work proposed in [7] (where the SS-Crowd component was described) is, to our knowledge, one of the first steps towards Conversing Learning. In that paper, the authors bring many interesting contributions to the idea proposed in our work. Thus, we've based the experiments (presented in Section 4) on a SS-Crowd implementation. In a nutshell, SS-Crowd takes (from a targeted learning system) potentially wrong knowledge, then, converts the specific knowledge into a question and query Yahoo!Answers community about the persistence of the question through their eyes. The answers from the community represent the belief that the knowledge from the learning system is right or wrong. SS-Crowd uses a predefined filter to combine the obtained answers and *decide* if the community answered in a positive or negative way. This *decision* can then be used to feedback NELL with information that indicates the differences between the knowledge acquired by the learning system with the knowledge from the web community. The Macro-Question/Answer approach is one of the key ideas behind SS-Crowd [7].

The idea of taking advantage on the redundancy of information from large content available on the web is focused in [8] to resolve QA problems. In that work, the amount of data available on-line makes answer extraction easier and the task presents a good performance even working on large datasets and simple natural language processing. Another interesting use of human generated content is presented on [9] the work applies frequently asked questions (FAQ) instead of traditional text files as a source to retrieve answers for a QA problem. Also, it introduces the FAQFinder system and has an approach to reduce the costs of natural language processing to understand complex questions. The system proposed matching the user's questions with existing questions on FAQ files.

In this paper we explore the usage of SS-Crowd in Twitter as well as in Yahoo!Answers communities. In addition we investigate the use of a supervised learning method to

[2] http://answers.yahoo.com/

[3] https://twitter.com/

learn to interpret the answers from both Web Communities. Twitter has been the focus of recent interesting researches. The work in [10] presents a network of stream-based measures called Tweetonomy that combines messages, users and content of messages to allow the measure to compare stream aggregations. Also, the work presented in [11] proposes a method to generate answers to status of Twitter users. Although the method is not intended to be a dialog machine, it succeeded in generating meaningful answers to the twitter statuses.

The collected intelligence as mentioned in [6], is the data retrieved from the social web and contains high valued information to web semantic development. Gruber suggests that the real collective intelligence comes from the creation of knowledge which is impossible to be obtained manually, and also from new ways of learning through the recombination of data from social web. Gruber describes the class of systems that can deliver at this opportunity as *collective knowledge systems* and he suggests four key properties that characterizes them. They are: user generated content, human-machine synergy, increasing results with scale and emergent knowledge.

3 Conversing Learning

A *Conversing Learning* (CL) system should be capable of autonomously looking for human collaboration to enhance a ML system. The collaboration can be used to perform self-supervision and self-reflection tasks. In this section, we define the behavior of a Conversing Learning system based on Web communities, its differences from Active and Interactive Learning and our concerns to improve communication among machines and users. Researchers in Human Computer Interaction (HCI), have worked on how users interact with machines, focusing mainly on making the user's experience more useful and friendly. In these cases, most of communication improvements targets the human users. In Conversing Learning instead, we want humans to help improving machine tasks, which means, the application of Reversed Human Computer Interaction (RHCI) [7]. In our approach, the communication improvements target the machine and not the human user. To autonomously improve Machine Learning tasks based on human supervision, we had to focus in an environment where the computer can autonomously get help from humans. Thus, it is important that a Conversing Learning system identifies the following questions: (i) which knowledge should be put to humans attention? (ii) Who are the humans that the machine should look for help? (iii) How to understand human answers? (iv) How to infer knowledge from human answers?

In this work, to demonstrate Conversing Learning capabilities, we explore new possibilities with the SS-Crowd algorithm. The algorithm was first presented on [7] and uses NELL' knowledge base (KB) to get together a machine that aims to learn as humans do and a machine that resolves its questions as humans do. SS-Crowd algorithm can be summarized by the following automatic tasks: (i) Take facts from the KB. (ii) Build a human understandable question from the facts. (iii) Query the web community with the questions. (iv) Gather and resolve the answers (classify them as positive or negative). (v) Combine the answers and produce a combined opinion from the community about the persistence of the facts. Although we expect Conversing Learning to bring some insights to enhance SS-Crowd capabilities, it is important to clarify that particular improvements on this system are not the focus of our research. Instead, we want to explore the possibilities and important issues of using Web communities in learning tasks aiming to present Conversing Learning as a series of new capabilities and concerns.

Popular AI applications like spam trackers already presented good solutions using systems based on Interactive Learning approaches. With machine learning techniques, a spam tracker may have its own set of initial rules (or facts) to find spam messages. The spam trackers apply Active Learning to select candidate messages to be labeled as spam by the users. The tracker keeps interactively asking the user and updates it's policy rules to identify new spams. In the spam tracker case (following the Interactive Learning approach) the e-mail owner is the only human that can interact with the machine, and the machine has no need (and no capability) to look for help anywhere else. Therefore, the machine prompts the user with questions and *passively* waits for collaboration.

An IL spam tracker depends on the user to complete its IL task. Thus, with no interaction with other humans and no *proactive* search for extra collaboration. In Conversing Learning, on the contrary, we want the system to *actively* look for help in other sources when needed. In addition, tagging e-mails may not be a long effort task for a regular user, but other Machine Learning tasks, such as Never Ending Learning might have a large set of data to be verified and the opinion of a single user may not be enough to feed the system accurately. A Conversing Learning system can share the validation task among several human users and use their different opinions as an advantage to provide redundancy. The core difference between Conversing Learning task to other learning tasks resides in (proactive and) automatically seeking for information from human users instead of passively waiting for their collaboration. Looking for help from many (and any) humans user may lead the IL task to lose precision and confidence due to noisy feedback. To rely on the human generated content, and answer the questions raised by Conversing Learning, we defined the following capabilities.

Active Learning approach: When the KB of the learning system is large, it might be unpractical to put every bit of knowledge to human validation, therefore it is necessary to prioritize the knowledge that is going to be validated by the web community. Also, querying the users for more information than it is usually done, will constrain the user to keep collaborating because they will not be able to track all the messages from the ML system. In Machine Learning we are used to actively select (from the dataset) the information that is more adequate to the ML system intents, that is, Active Learning. And this is what we are doing here, it is important to select the bit of knowledge that brings better benefits when asked to a web community.

Scope of the Web Users: The effectiveness of asking for human feedback can be different from a community to another. The web communities available online have different users and different intents. Although we can communicate with users in almost any community, they might react in a different way. The culture, expertise, age and language are just a few factors that will change our feedback message. For an example, a travel suggestion application that reads from a KB would be better fed from users of travel web communities than an open question answering community like Yahoo! Answers, but you would have to deal with the drawback of querying a smaller set of users with the risk of collecting a smaller set of feedback.

Driven feedback: When we are working with human generated content, the answers (gotten from human users) might be too noisy or too complex to be interpreted. If that is the case, the algorithm can miss part of the feedback. An alternative to cope with such cases is to encourage human users to provide *machine friendly* content. A good example of such an idea is implemented in [7] where the SS-Crowd algorithm prompts Yahoo! Answers users to answer just *yes* or *no* for their questions. Such approach restricts the answers from the users and the algorithm can be structured to focus on content easier to read. As in the case of scope, the drawback of such approach is to find a smaller amount of contributions. Although we consider this is imperative to enable accurate results in Conversing Learning, the recent advances in Natural Language Processing (NLP) and IR indicates that machines will be able to better and better understand human generated content in the next years, and soon, we would not need to push the user to an specific kind of answer.

System Identity: It is known that the human communication behavior changes as the interlocutor changes. If you target web users that know about the machine nature of who is asking, they might feel either cornered and shy or stimulated when returning feedback. Researchers have already been doing it through Amazon's Mechanical Turk [4], where users are stimulated with specific instructions to feedback a system they know it is as a machine. If the community is used to help academic research, it is likely that the feedback will be more *machine friendly*. Showing ourselves as a machine, or not, depend on the intents of the application of Conversing Learning but in either way, it is important to know the aspects of the community we are asking for feedback and to ensure that this usage of the social media does not bypass its security and privacy policies.

4 Experiments and Results

To explore Conversing Learning principles, and also, to study the interaction of different web communities, we ran the SS-Crowd algorithm using Twitter as well as Yahoo!Answers as a source for human feedback. Although Twitter interface is not intended to perform as a QA system, users often use it to get answers for *question posts*, so we are miming these users and behaving the same way. Considering that the work in [7] already put the SS-Crowd algorithm to test Yahoo! Answers, we are using here the same algorithm and adding Twitter as a second source of information.

With the method proposed in this paper, we want to explore how can we apply Conversing Learning by implementing its capabilities in a real case where NELL would benefit from the obtained results. We are also going to explore how the behavior of different communities could affect the benefits of using social media as a source of information for learning tasks. As a measure of achievement, we took the very same rules used in [7]. We had a set of 62 NELL's rules that were (automatically) converted (by SS-Crowd) into questions and then, were posted as questions in both communities. The questions generated 350 answers in Yahoo! Answers and 72 answers in Twitter. All those results were given as input to a classifier to learn how to interpret answers from

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

both communities. In our experiments each question receives several kinds of sentences as answers and the SS-Crowd algorithm determines if those answers are approving or rejecting the the validity of the rule. If the algorithm cannot make such decision then the rule is marked as unresolved.

During the execution of the experiments, we noticed how user's collaboration differ from one community to another. In Yahoo! Answers, users are not aware that our questions are generated by a machine. In Twitter instead, we made that clear. As an instant effect, users came to us asking how restrictive should they be about the rule being evaluated. We answered them to be as restrictive as they think they should be, so we keep our intents to not interfere in the user's opinion. Knowing the original intents of the question, the users in Twitter are more *machine friendly*, they actually try to help the learning system (considering that they are NELL's *followers*).

In Yahoo! Answers, people are encouraged to earn *points* and *respect* by answering questions. Users collaborate giving answers even when they are not sure about the answer. This behavior reflects in our results as a higher amount of collaboration. In Twitter instead, the collaboration has some restrictions. The user has to be *following* NELL to receive its updates (questions in our case). This means that the user is previously interested in the subject and because of this interest, while the amount of collaboration decreases, its quality increases a lot. The example below extracted from our results explains it in a practical manner.

Question: ***(Yes or No?) If athlete Z is member of team X and athlete Z plays in league Y, then team X plays in league Y.***

Answer sampled from a Twitter user: ***No. (Z in X)*** $\wedge$ ***(Z in Y)*** $\rightarrow$ ***(X in Y)***

Answer sampled from a Yahoo! Answers users: ***NO, Not in EVERY case. Athlete Z could be a member of football team X and he could also play in his pub's Friday nights dart team. The Dart team could play in league Y (and Z therefore by definition plays in league Y). This does not mean that the football team plays in the darts league!***

As we can infer from the examples, users from both communities are giving us the same opinion through different answers. The first contains a simple *No* answer and a justification in a logic-like format while, the latter, is pure natural language and includes an example. Everything that we need from both answers is the *No* and since the first answer is shorter, the SS-Crowd is more accurate to extract the opinion from it. In table 1, we notice more unresolved answers in Yahoo! Answers (16.5%) answers than in Twitter (5.5%) answers. It is also important to notice that the Yes/No nature of the question facilitates the resolution of tasks like this. This feature is the *driven feedback* discussed in Section 3 and is part of the SS-Crowd original algorithm. Overall, as illustrated in the example, we can state that if the users are different, the system is different and the answers are different.

	Approved	Rejected	Unresolved
Twitter	51	17	4
Yahoo! Answers	124	168	58

Table 1. Total of approved, rejected and unresolved answers from Yahoo! Answers and Twitter

The Conversing Learning system should be able to find how useful is the community contribution. If the human collaboration is not good enough, the system may take an action to help users to provide better feedback. This interaction between the learning system and the users aiming to allow human feedback in machine learning tasks is the main focus of Conversing Learning.

We know that the *machine friendly* collaboration of Twitter users is good to our intents since it allows more accurate validation of knowledge. We also know that a QA environment such as Yahoo! Answers is more participatory and we have users from all kind of expertise and experience. Thus, we have on one side a more accurate and smaller set of answers and in the other a larger set of answers, human *identity* and an unbiased crowd. Those different biases present in each community were important to help us deciding upon using these specific communities. Examining the simple sum of the totals of answers from both communities, we found that users from Yahoo! Answers and Twitter have a substantial difference in their opinion, which is good to our intents. The results also pointed that users from one community disagree with users from the other community in 45% of the answers. This increases our belief that we are not dealing with redundant information (but with independent sources).

A Conversing Learning system with multiple independent sources of human collaboration could use collective knowledge to improve its own ability to keep looking for information. Therefore, performing a self-revision task. To implement such capability, we gathered information from SS-Crowd implementation with Twitter and Yahoo! Answers and represented the data as attributes to a classifier. Thus, the system is capable of assisting SS-Crowd to identify where to look for better information on web communities. The attributes retrieved from SS-Crowd are as follows: (i) Total number of rules resolved as approved and rejected by users in Twitter. (ii) Total number of rules resolved as approved and rejected by users in Yahoo! Answers. (iii) The best answer from Yahoo! Answer. (iv) The combined resolution of answers to a single question (taken from SS-Crowd task #5 in Section 3).

We believe that the classifier can be used to infer more valuable knowledge from the behavior of the communities. If we are applying Conversing Learning to improve learning system KB, such a classifier could bring the possibility to choose what information from the web community makes difference to that specific learning system. In our experiments we used the attributes to feed traditional classifiers and observed how the combination of different social media sources and the improved interaction with users, through Conversational Learning, could give us a deeper understanding of the machine knowledge validated through the eyes of humans.

To apply the Conversing Learning idea in our experiments, we used the attributes retrieved from SS-Crowd to create 62 tuples (one for each rule) to train traditional classifiers. The learning task of the binary classifier is to identify whether a rule is right or wrong. The dataset composed by those instances (tuples) were previously labeled with the judgment of NELL's developers and the tests were performed using a 10-fold cross-validation. With the outputs, we can measure the relevance of the attributes and decide which of them are suitable to our intents. Although the average difference indicates no redundant information, it does not guarantee that every attribute are not independent. We know that some attributes may represent our problem better than others, and to resolve this matter, we decided to run the classifiers in a ablation strategy removing attributes from the dataset. Therefore, we are going to analyze how the classifier accuracy behaves in the lack of attributes.

	Classifier		
Removed Attribute	NaiveBayes	C4.5	ID3
None Removed	74.19	77.41	75.80
YahooApproved	75.80	74.19	75.80
YahooRejected	**69.35**	**69.35**	**72.58**
YahooUnresolved	75.80	79.03	80.64
TwitterApproved	75.80	61.29	61.29
TwitterRejected	74.19	83.87	77.41
TwitterUnesolved	70.96	75.80	75.80
YahooBest	77.41	75.80	77.41
YahooCombined	74.19	75.80	75.80
YahooOnly	70.96	77.41	74.19
TwitterOnly	**77.41**	**79.03**	**79.03**

Table 2. Classifier average correct classification rates over 10-fold cross validation using a dataset containing 62 instances

When comparing both systems' outputs, it is possible to notice that Yahoo! Answers attributes are more relevant to the Conversing Learning system. The system could benefit from this inference to assign a different behavior for Yahoo! Answers collaboration, specially for the rejections, as evidenced in Table 2. On the other hand, when using a single community as source of human feedback, the use of Twitter brought better results than Yahoo!Answers (last two lines in Table 2). As we already mentioned, the *identity* of the Conversing Learning system and the knowledge of the humans about a subject might interfere in their decisions. While Twitter users gives us more straightforward response (which matches the NELL's developers), the Yahoo! Answers users gives more complex and more in-depth feedback. In a few words, while Twitter users improves a ML in self-supervision, Yahoo! Answer users help the ML system to ensure its completeness, that is, the system comprehension of all possibilities of the knowledge acquired.

In a nutshell, the results of our classifier can tell which attributes are more relevant to the rule validation task. The classifier loses accuracy when the YahooRejected attribute is removed. Implementing such a classifier can help the Conversing Learning system to tune itself in a self-supervised self-reflection task and to be able to be more effective. The classification task can also report to the ML system, which bit of the knowledge base could benefit from deeper investigation.

5 Conclusions and Future Work

In this work, Conversing Learning was proposed and implemented using NELL as a case study. Such a learning process is intended to autonomously help to improve

ML tasks actively looking for human assistance from different sources (in a *Active-Learning-oriented approach*. To achieve that, the presented case study took advantage from the web communities and their wide popularity and also their millions of users. We presented our concerns and some directions on how to solve problems of low confidence when relying on human generated content. We also showed how Conversing Learning systems are related to Active Learning and Interactive Learning. To allow Conversing Learning systems to effectively communicate with humans, we explored some ”reversed” techniques such as Reversed Human Computer Interaction as well as Reversed Macro Question/Answeras defined in [7].

The case study was implemented based on SS-Crowd and three traditional classifiers (Naive-Bayes, C4.5 and ID3). Experiments were performed using Twitter and Yahoo!Answers as source for the human feedback. The results obtained in the performed experiments revealed that the Conversing Learning approach was able to correctly label data that can be stored in NELL's knowledge base and help the system in its never-ending learning task. Also, it was possible to observe that Twitter and Yahoo!Answers classifier attributes contributed in different ways to the classification task. In this sense, Twitter's attributes alone were capable of giving the higher classification accuracy to the classifiers. However, the list of used attributes can still be extended to take advantage of other information from web communities such as reputation and earned points.

Since the SS-Crowd algorithm has a few issues with unresolved answers (e.g. decide if an answer is *yes* or *no*), an interesting future work can focus on extending the keyword base approach [7] with more detailed information from the web community (like opinion analysis and sentiment analysis). We also intend to increase the datasets used in the experiments to learn more subtleties from Conversing Learning approach.

References

1. Cohn, D.A., Ghahramani, Z., Jordan, M.I.: Active learning with statistical models. Journal of Artificial Intelligence Research **4** (1996) 129–145
2. Settles, B.: Active Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan and Claypool (2012)
3. Godbole, S., Harpale, A., Sarawagi, S., Chakrabarti, S.: Document classification through interactive supervision of document and term labels. In: In PKDD-04. (2004) 185–196
4. Settles, B.: Closing the loop: Fast, interactive semi-supervised annotation with queries on features and instances. In: Proc. of the EMNLP'11, Edinburgh, ACL (2011) 1467–1478
5. Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Jr., E.R.H., Mitchell, T.M.: Toward an architecture for never-ending language learning. In: Proceedings of the Twenty-Fourth Conference on Artificial Intelligence (AAAI 2010). (2010)
6. Gruber, T.: Collective knowledge systems: Where the social web meets the semantic web. Web Semantics **6**(1) (February 2008) 4–13
7. Pedro, S.D.S., Hruschka Jr., E.R.: Collective intelligence as a source for machine learning self-supervision. In: Proc. of the 4th International Workshop on Web Intelligence and Communities. WIC12, NY, USA, ACM (2012) 5:1–5:9
8. Dumais, S., Banko, M., Brill, E., Lin, J., Ng, A.: Web question answering: is more always better? In: Proc. of the 25th annual ACM SIGIR, NY, USA, ACM (2002) 291–298
9. Burke, R.D., Hammond, K.J., Kulyukin, V.A., Lytinen, S.L., Tomuro, N., Schoenberg, S.: Question answering from frequently asked question files: Experiences with the faq finder system. Technical report, Chicago, IL, USA (1997)

10. Wagner, C., Strohmaier, M.: The wisdom in tweetonomies: acquiring latent conceptual structures from social awareness streams. In: Proc. of the 3rd Int. Semantic Search Workshop, New York, NY, USA, ACM (2010) 6:1–6:10
11. Ritter, A., Cherry, C., Dolan, W.B.: Data-driven response generation in social media. In: Proc. of the. EMNLP '11, Edinburgh, UK, ACL (2011) 583–593