Towards the Machine Comprehension of Text: An Essay

Christopher J.C. Burges
Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA

December 23, 2013

Technical Report
MSR-TR-2013-125

Microsoft Research
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
http://www.research.microsoft.com

1 Introduction

The Machine Comprehension of Text (MCT)[1] has been a central goal of Artificial Intelligence for over fifty years. How does one even define "machine comprehension"? Researchers often invoke the Turing test to this end (a machine attains human level intelligence if its responses in a dialog with a human are indistinguishable from those of another human (Turing, 1950)), but as Levesque (2013) recently pointed out, this definition has resulted in workers focusing on the wrong task, namely, fooling humans, rather than achieving machine intelligence. But even if researchers could be persuaded to focus on the AI part of the Turing test, the test is still a false goal, in the sense that the typical user would be happy to know that she is having a dialog with a machine if this were a result of her knowing that no human could possibly be that smart. Perhaps shoehorning the research to meet the goal of appearing human-like is a red herring. Levesque also suggests multiple choice tests that require world knowledge (for example, to solve the anaphora problem) as a suitable replacement for the Turing test. We will return to multiple choice tests below. But this still leaves the definition of machine comprehension tied to the data used to construct the tests. It seems useful to define the task more generally, but still operationally, and to this end we suggest the following: *A machine comprehends a passage of text if, for any question regarding that text that can be answered correctly by a majority of native speakers, that machine can provide a string which those speakers would agree both answers that question, and does not contain information irrelevant to that question.* Thus we can define machine comprehension in terms of Question Answering in its most general form. Much has changed since the early days and we can hope that recent advances, such as the emergence of large, distantly labeled datasets (e.g. text on the Web), the availability of orders of magnitude more computing power, and the development of powerful and principled mathematical models, will lead to real progress. The goal of this essay is to examine what might be needed to solve the problem of the machine comprehension of text.

[1] We prefer this term as more precise than other terms such as Machine Reading (but machines have been reading since the days of punch cards) and Natural Language Understanding (which, as a challenge, can equally apply to people).

1.1 How To Measure Progress

Levesque (2013) suggests multiple choice question answering as a better alternative to the Turing test. To spur research in this direction we have made available a dataset of 660 fictional short stories, created using crowd sourcing, and aimed at the reading level of a typical 7 year old (Richardson et al., 2013). Each story is accompanied by four multiple choice questions. The Winograd Schema Test proposal (Levesque, 2013) suggests using questions that require significant expertise to generate, since the question/answer pairs are carefully designed to require background knowledge (for example, in *"The ball fell through the table because it was made of paper"*, to what does "it" refer?"). On the other hand, using crowd sourcing to generate the data has the significant advantage of scalability. We also have some control over the difficulty of the task by restricting the available vocabulary. If progress is rapid and the data set turns out to be too easy, we can increase the vocabulary from the current 8000 words, incorporate non-fictional writing, and if necessary, change the task definition, by, for example, (1) not requiring that exactly one of the four alternative answers always be correct, but instead allowing more than one, one, or no correct answers per question, or (2) requiring that more answers require reasoning over several sentences (for the current set, workers were asked to make at least two questions answerable only by combining information from at least two sentences; this could also be tightened by requiring that the two sentences be separated in the text). It is interesting that, while random guessing will get 25% of the questions correct, a simple token-based baseline achieves approximately 60% correct, and early results using a modern textual entailment system are similar (Richardson et al., 2013).

2 Desiderata and some Recent Work

Writing down a set of desiderata for an adaptive machine comprehension system rapidly leads to a set of interesting, unsolved problems:

1. The meaning representation should be scal-

able in two different senses, namely that (1) it is learned in an almost entirely unsupervised fashion, leveraging widely different sources of often noisy data, and (2) it can be applied to arbitrary problem domains.

2. Inference should be accomplished in real time, no matter how large and complex the model becomes.

3. For ease of construction and debugging, the system should be built using composable modules. The modularity should enable the system to react efficiently (i.e. be able to learn in real time) although continuous background learning involving the whole system may also be necessary.

4. Each module should be interpretable, and compositions of modules should also be interpretable. That is, given that the overall system makes an error, it should be possible to understand why it makes that error.

5. The system should be monotonically correctable. That is, given that it makes an error, it should be clear how to alter the model to correct that error, without introducing new errors.

6. The system must be able to perform logical inference over its meaning representation: that is, it should be able to draw logical conclusions by combining its world model with its representation of the text. It seems likely that such reasoning should be probabilistic, to avoid brittleness and to combine beliefs correctly (Richardson and Domingos, 2006).

7. The system should be interrogable: that is, it should be able to make an assertion as to the (probability of) the truth of any Boolean hypothesis presented to it, and through its interpretability, we should be able to understand why it makes this assertion.

Let's examine these in turn. Current work on modeling meaning often requires expensively annotated data, such as tagging sentences with their logical forms (Zettlemoyer and Collins, 2005; Zettlemoyer and Collins, 2009; Kwiatkowski et al., 2010). An interesting intermediate approach is to induce the logical forms from much cheaper annotations such as question-answer pairs (Liang et al., 2011). Some methods for achieving scalability through unsupervised learning have been proposed: Goldwasser et al. (2011) use self training to achieve 60% accuracy on the Geoquery data (Zelle and Mooney, 1996) (as opposed to the 80% attained through supervised training) and Poon and Domingos (2009) rely on clustering semantic text segments. It's possible that data scalability could also be achieved using crowd sourcing, especially if the task can be designed as a game (Ahn and Dabbish, 2004) - could millions of people be enticed to help, with the goal of helping an artificial child grow and learn English? But the availability of huge, inexpensive datasets, and recent findings that some level of semantic modeling can be achieved using large amounts of unlabeled data (Mikolov et al., 2013), suggest that we would be wise to leverage such resources. One recurring problem in semantic modeling has been that meaning representations designed for one task tend not to be portable to another; our design should attempt to avoid this problem from the start. Humans achieve (close to) real time inference using (many) processors that are orders of magnitude slower than the chip in your phone, so real time inference seems like a reasonable, and very useful, expectation. Bottou (2011) has emphasized the need for composable modules. It is too easy, when attacking a very challenging task, to lose one's way, and modularity, debuggability and interpretability should help prevent this (it is no coincidence that the currect record holder for the Imagenet classification competition designed methods to understand what their convolutional nets are actually doing: (Zeiler and Fergus, 2013)). Correctability is closely related: current machine learning models generally give no guarantees on not making new errors when correcting the old; decision surfaces can move in unpredicatble ways when the the training data changes. Thus correctability would also give us more control over system behaviour. It is clear that logical inference is needed for comprehension at least some of the time (although perhaps less than we think). Interrogability is another debugging tool: we are simply more likely to succeed if we have models we can more easily understand.

3 Seven Signposts

3.1 How to Incorporate Structure in Learning?

One can view early approaches to Artificial Intelligence (AI) and modern approaches to Machine Learning (ML) as extremes lying on a spectrum of methods. Early AI used rules and ontologies, but these were heuristic, brittle, and non-scalable. Although ML approaches have been extended in various ways to handle structured data (see, for example, Tsochantaridis et al. (2004); Taskar et al. (2005); Ganchev et al. (2010); Chang et al. (2012)), the main underlying approach is predominantly statistical: the ML supervised setting usually requires that instances be drawn from a fixed distribution and then considers a family of models, a cost function, and a labeled dataset, and the goal is to find that model for which the error measure at hand (to which the cost is often a convex approximation) is minimized on out-of-sample data[2]. Structure is then imposed from within this framework, often by limiting the search over functions to those that satisfy the structural constraints. But langauge is extraordinarily structured, and it seems that an approach that recognizes this from the start, rather than tweaking ML models to incorporate structure, will have better chances. Put another way, ML can be viewed as a principled way to handle our uncertainty. We must resist the temptation to model the uncertainty in the data too soon, if it leads to our ignoring much of what we know about the structure in the data; and most machine learning algorithms are exquisitely tuned to model uncertainty, using extraordinarily simple labels (for example, binary classes) as their signal. Probabilistic graphical models do address the problem of stucture in the model itself, but that structure must usually be hand designed and thus is not scalable, such models tend to quickly become intractable with size, and they only encode dependency structures, whereas it seems likely that the many other kinds of structure inherent in text need to be built into the model from the ground up. As described above, recent work combines logical structure with statistical modeling more directly, but scalability remains a central problem; and in the large, we have gone from one extreme (early AI, with hand-built rules) to another (modern ML, with no rules at all).

[2]To simplify we omit the role of regularization here, which can be viewed as a choice of prior on the function class.

3.2 Do Large Data and Deep Learning Hold the Key?

Over the last decade it has been shown that, given enough data, remarkable progress can be made on traditionally very hard tasks like Question-Answering, ontology building, and unsupervised image modeling. The AskMSR Question-Answering system can answer complex questions using only hand-designed pattern matching coupled with web search (Brill et al., 2002). When labeled data sets increase by orders of magnitude, different methods can converge in accuracy (Banko and Brill, 2001), which suggests that access to large datasets can be more important than choice of algorithm[3]. More recently, the Never Ending Language Learning (NELL) system crawls the web continuously, developing an ever increasing set of categories and relations in a mostly unsupervised manner (Carlson et al., 2010). Recent work training a deep neural network on images from YouTube videos showed that "grandmother neurons" can occur naturally in artificial systems (Le et al., 2012). However, systems that avoid semantic modeling and that instead rely on the scale of the data can be brittle. When asked *How many feet are there in a lightyear?*, the AskMSR system answered *Winnie the Pooh*, through a chain of reasoning that involved Buzz Lightyear, a Disney character[4]. Problems like this can be addressed with shallow semantic processing (such as requiring that, for questions that begin with the adjectival phrase *how many*, the answer be a number), but we have no reason to believe that such techniques will fully solve the problem. Similar considerations apply to the Watson system (Ferrucci et al., 2010), which applied ensembles of information retrieval techniques to achieve a significant milestone in IR, but was nevertheless brittle (e.g. in response to a question asking for a city in the US, Watson gave the response *Toronto*). Deep learning is a powerful paradigm, although many of the ideas have been around for decades, and deep convolutional nets shown to work well long ago (LeCun et

[3]Although note that the task considered in (Banko and Brill, 2001) was shallow and would clearly benefit from large datasets.

[4]"light year" should have been two words, in the question.

al., 1998); but even deep learning's recent successes in speech (see for example (Yu et al., 2013)) and image classification ((Zeiler and Fergus, 2013)) rely on systems that are not interpretable, interrogable, modular, composable, correctable, or scalable, in the senses given above. It seems that such successful adaptive systems will provide valuable components, but that much more will be needed to solve "strong AI".

3.3 Why is NLP so Hard?

Natural Language Processing, which can be viewed as another approach to directly modeling structure in text, is still in its infancy: problems that seem simple to humans, such as whether a sentence makes sense or not, or even whether or not it is grammatically correct, are as yet unsolved. One reason seems to be that understanding natural language, for a given domain, requires having access to a rich model of the world, for that domain; but building such rich models requires the ability to process natural language. This has led some researchers to frame their work by limiting the problem scope to tasks for which the world model is more accessible (Chen and Mooney, 2011; Zelle and Mooney, 1996; Tang and Mooney, 2001). Perhaps another reason is that training machine learning models on NLP tasks tends to focus attention on problems that require very simple labels (such as binary classes) rather than trying to directly leverage the rich structure in the data.

3.4 Can we Limit Scope for Manageability, yet Still Achieve Scalability?

Modern attempts to model meaning often limit the scope of the problem in order to make it tractable. But limiting the scope can result in solutions that are not scalable. The ATIS data has been used for two decades as a resource for studying the mapping of spoken language to intent, and for slot filling, in the air travel reservations domain (see (Tur et al., 2010) for a recent review) yet the general problem (of mapping language to intent) remains unsolved. The same data has been used to show that sentences can be mapped to first order logical forms using training data consisting of sentences annotated with lambda-calculus logical forms, but creating such training data is expensive and therefore non-scalable (Zettlemoyer and Collins, 2009). Similar methods have been applied to other limited scope problems (for example, question answering over geographical data (Kwiatkowski et al., 2010)). There is clearly scientific value in considering limited scope problems, but somehow we must find problems with a particular kind of scope limitation that allows easy generalization to new domains and to larger datasets. It was this observation that led us to propose using a limited vocabulary grade level multiple choice reading comprehension task (Richardson et al., 2013); the problem is open domain, but has limited scope, and the task can be easily modified to raise the bar for state of the art methods, as those methods improve.

3.5 How to Ground Meaning?

Attempting to model meaning using text alone makes the problem much harder. For example, dictionary definitions can often be circular: Wiktionary defines *descendant* as *one who is the progeny of someone* and it defines *progeny* as *offspring or descendants*. Recent work has attempted to tackle both the scope and the grounding problems at the same time: for example, learning to navigate a maze using natural language instructions (Chen and Mooney, 2011), or learning to sportscast for a RoboCup soccer game by observing how humans do it (Chen and Mooney, 2008). The discipline that searches for the most compact possible set of underlying rules that describe the physical world, is physics. Since hundreds of years of effort have been applied to finding such rules, it seems wise to leverage them where appropriate: thus perhaps concepts like space, time, mass and energy should play a central role in any general world model. The extraordinary sparseness of the rules of physics, when compared to the complexity of the world they describe, would help make a world model built using them both concise and powerful, when those rules apply. However, physics does not describe complex phenomena (such as people). This leads to the question: is it possible to craft a fundamental set of "ground rules" that describe the basic properties of components in the world model (such as sentient entities), in such a way that those rules can be extended, scalably and largely automatically, as needed? That is, can we solve the grounding problem by starting with a carefully designed template for the world model? We must avoid brittle, non-scalable rule based systems: but brittleness

and non-scalability would be avoided if we could design a system whose very sparse, template ground rules could be expanded using unlabeled (or cheaply labeled) data.

3.6 Are Brains Using Machine Learning?

Suppose that you discover that your friend is laboring under a misunderstanding. How do you remedy this? You don't lock him in a room with terabytes of training data and ask him to spend a week updating his parameters. You, within a matter of seconds, using a processor whose time scales are of order tens of milliseconds, form a model in your own mind of what his misunderstanding is, the most likely way in which he arrived at it, and what exact beliefs in his world model need updating; that is, you have an interpretable model of his beliefs. You may need, along the way, to ask your friend one or two questions: he is interrogable. Modularity at some level is suggested by the strong compartmentalization of human learning: when people learn to ride a bicycle, they don't forget how to brush their teeth. In fact, they don't appear to forget anything else. This is in stark contrast to a typical machine-learned model, where retraining with additional data will (hopefully) move the decision surface in such a way that the error rate on unseen data is reduced, but will also typically result in some instances that were previously classified correctly becoming errors.

It seems to work this way no matter what the misunderstanding is. Humans never run into having to use large, labeled training datasets when discerning the meaning of something. Even if the task is very complex (e.g. enlarging our understanding by taking a course), the learning process is broken into small steps of updating our models of the world, rather than blind, statistical parameter adjustment. People must have a model of meaning that they can easily both poll and update, which suggests that to them at least, their model is interpretable, correctable and interrogable. It may well be that we are using extremely complex, intermediate meaning representations that we could not understand even if we could map them out (in terms of synaptic connection strengths, neuron firing rates, etc.); but the end effect is that meaning is extremely accessible to us. This is an extraordinary fact, and it elicits the same feeling of awe, in me at least, as the fact that we are able to model the world mathematically, at all. But rather than trying to solve the problem by modeling how humans do it, perhaps rich world models could more directly enable the scalable, modular, composable, interpretable, interrogable and correctable models we desire.

3.7 Can we Leverage the Many-to-One Mapping of Text to Meaning?

Language is highly redundant, yet largely unidirectional: although there are many ways of saying the same thing, the reverse - statements that have several possible meanings to the recipient - are rare, presumably because ambiguity in meaning is usually counterproductive to communication. Of course the story is different for text whose intended purpose lies elsewhere (for example, jokes, or poems). Paraphrase detection and generation are known to be very hard problems. But this many-to-one mapping suggests an opportunity: if a model encounters a text which amounts to a different way of saying something it already "knows", and if the number of things that that text could possibly mean is limited, then perhaps the model could infer that the new statement maps to the same meaning representation, with no need for further labeling.

4 Discussion

Rule-building has been around since the beginning of AI, and researchers are understandably wary about relying too heavily upon it. The main reason that expert systems did not solve AI is that with rare exceptions, updating rules manually is not a feasible large scale solution. However, techniques that require significant up front manual labor, but which are then largely automated, can be scalable, as NELL has shown; perhaps some form of scalable rule building and rule learning can be made to work. The use of machine learning, although clearly very powerful in many scenarios, can also act as a brake on progress, since such systems are typically not interpretable and so are difficult to further improve, without yet more labeled data. They also currently lack the other key properties we describe above. However clearly such powerful tools should be used when appropriate. One approach may be to limit their use to model the uncertainty

remaining in the data after all structure in the data has been maximally leveraged. The fact that text is usually unambiguous to humans also suggests that for people at least, the modeling of uncertainty is not the central problem being solved. Saving our current machine learning algorithms (deep or otherwise) for those situations where uncertainty must be modeled and where labels are extremely simple, and instead searching for different approaches designed for richly structured yet unambiguous text, is not meant to suggest that we abandon the rich mathematical foundations upon which machine learning rests; the same search for disciplined methods should serve us well here, too.

Acknowledgments

I wish to thank Hoifung Poon, Peter Bailey, Dan Schwartz, Moises Goldszmidt, Matt Richardson, Scott Yih, John Platt, Andrzej Pastusiak, Erin Renshaw, Bill Dolan, Chris Brockett, and Josh Tenenbaum for valuable discussions.

References

L. Von Ahn and L. Dabbish. 2004. Labeling images with a computer game. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 319–326. ACM.

Michele Banko and Eric Brill. 2001. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 26–33. Association for Computational Linguistics.

Léon Bottou. 2011. From Machine Learning to Machine Reasoning. *CoRR*, abs/1102.1808.

E. Brill, S. Dumais, and M. Banko. 2002. An analysis of the askmsr question-answering system. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E.R. Hruschka Jr., and T.M. Mitchell. 2010. Toward an architecture for never-ending language learning. In *Proceedings of the Conference on Artificial Intelligence (AAAI), 2010*.

M. Chang, L. Ratinov, and D. Roth. 2012. Structured learning with constrained conditional models. *Machine Learning*, 88(3):399–431, 6.

D. Chen and R.J. Mooney. 2008. Learning to sportscast: a test of grounded language acquisition. In *Proceedings of the 25th International Conference on Machine Learning*, pages 128–135. ACM.

D.L. Chen and R.J. Mooney. 2011. Learning to Interpret Natural Language Navigation Instructions from Observations. In *AAAI*.

D. Ferrucci, E. Brown, J. Chu-Carroll, J. Fan, D. Gondek, A.A. Kalyanpur, A. Lally, J.W. Murdock, E. Nyberg, and J. Prager. 2010. Building Watson: An overview of the DeepQA project. *AI magazine*, 31(3):59–79.

K. Ganchev, J. Graca, J. Gillenwater, and B. Taskar. 2010. Posterior Regularization for Structured Latent Variable Models. *JMLR*, 11.

D. Goldwasser, R. Reichart, J. Clarke, and D. Roth. 2011. Confidence driven unsupervised semantic parsing. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 1486–1495. The Association for Computer Linguistics.

T. Kwiatkowski, L. Zettlemoyer, S. Goldwater, and M. Steedman. 2010. Inducing probabilistic ccg grammars from logical form with higher-order unification. In *Proceedings of the 2010 conference on empirical methods in natural language processing*, pages 1223–1233. Association for Computational Linguistics.

Q.V. Le, M. Ranzato, R. Monga, M. Devin, K. Chen, G.S. Corrado, J. Dean, and A.Y. Ng. 2012. Building high-level features using large scale unsupervised learning. In *ICML*.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Hector J. Levesque. 2013. On our best behavior. In *International Joint Conference on Artificial Intelligence*.

P. Liang, M. Jordan, and D. Klein. 2011. Learning dependency-based compositional semantics. In *ACL*.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Hoifung Poon and Pedro Domingos. 2009. Unsupervised semantic parsing. In *EMNLP*, pages 1–10.

M. Richardson and P. Domingos. 2006. Markov logic networks. *Machine learning*, 62(1-2):107–136.

M. Richardson, C.J.C. Burges, and E. Renshaw. 2013. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *Empirical Methods in Natural Language Processing (EMNLP)*.

L.R. Tang and R.J. Mooney. 2001. Using multiple clause constructors in inductive logic programming for semantic parsing. In *Proceedings of the 12th European Conference on Machine Learning (ECML)*, pages 466–477, Freiburg, Germany.

B. Taskar, V. Chatalbashev, D. Koller, and C. Guestrin. 2005. Learning structured prediction models: A large margin approach. In *ICML 22*, Bonn, Germany.

I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. 2004. Support vector machine learning for interdependent and structured output spaces. In *ICML 24*.

G. Tur, D. Hakkani-Tur, and L. Heck. 2010. What is left to be understood in atis? In *Spoken Language Technology Workshop*, pages 19–24. IEEE.

A.M. Turing. 1950. Computing machinery and intelligence. *Mind*, 59(236):433–460.

D. Yu, L. Deng, and F. Seide. 2013. The deep tensor neural network with applications to large vocabulary speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(2):388–396.

M.D. Zeiler and R. Fergus. 2013. Visualizing and understanding convolutional networks. *arXiv preprint arXiv:1311.2901 v3*.

J.M. Zelle and R.J. Mooney. 1996. Learning to parse database queries using inductive logic programming. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI)*, pages 1050–1055.

L. Zettlemoyer and M. Collins. 2005. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *Proceedings of the Twenty First Conference on Uncertainty in Artificial Intelligence (UAI)*.

L.S. Zettlemoyer and M. Collins. 2009. Learning context-dependent mappings from sentences to logical form. In *ACL/AFNLP*, pages 976–984. The Association for Computer Linguistics.