Towards a structured evaluation of improv-bots: Improvisational theatre as a non-goal-driven dialogue system

Maria Skeppstedt[1,2], **Magnus Ahltorp**[3]
[1] Computer Science Department, Linnaeus University, Växjö, Sweden
[2] Applied Computational Linguistics, University of Potsdam, Potsdam, Germany
[3] Magnus Ahltorp Datakonsult, Stockholm, Sweden
maria.skeppstedt@lnu.se, magnus@ahltorpdata.se

Abstract

We have here suggested a structured procedure for evaluating artificially produced improvisational theatre dialogue. We have, in addition, provided some examples of dialogues generated within the evaluation framework suggested. Although the end goal of a bot that produces improvisational theatre should be to perform against human actors, we consider the task of having two improv-bots perform against each other as a setting for which it is easier to carry out a reliable evaluation. To better approximate the end goal of having two independent entities that act against each other, we suggest that these two bots should not be allowed to be trained on the same training data. In addition, we suggest the use of the two initial dialogue lines from human-written dialogues as input for the artificially generated scenes, as well as to use the same human-written dialogues in the evaluation procedure for the artificially generated theatre dialogues.

1 Introduction

Improvisational theatre (or impro/improv) is an art form in which unscripted theatre is performed. Dialogue, characters and actions are typically created spontaneously. Through collaboratively creating a story, the actors can make a new scene evolve in front of the audience [Wikipedia contributors, 2018].

Seen from the perspective of artificial intelligence research, improvisational theatre is a sub-genre of human interaction that is more forgiving than interaction in general. Errors made in general interaction are typically seen as a failure, and in the case of a dialogue system, errors might lead to a dialogue breakdown. In contrast, errors made within an improvisational theatre scene are encouraged, and can form an input to how the scene evolves. It might, therefore, be interesting to find out how artificially constructed improvisational theatre bots, which are likely to make errors to a larger extent than a human, are perceived in this special setting.

Although there is previous work on the construction of artificially generated improvisational theatre, there are, to the best of our knowledge, no descriptions of structured methods for the evaluation of the dialogues created, and thereby no method for comparing different approaches for dialogue generation. According to Serban et al. [2016], even the more general question of which evaluation method to use for non-goal-driven dialogue systems (for which improvisational theatre could be claimed to be a sub-category), is an open one.

The aim of this paper is therefore to i) provide a suggestion for a structured procedure for evaluating artificially produced improvisational theatre dialogue, and ii) give some examples of dialogues generated within the evaluation framework suggested.

2 Previous work

Creating artificially generated human dialogue is a classical task within the research field of artificial intelligence, with the ultimate aim of a bot being able to pass the Turing test. Dialogue could either be created in the form of a goal-driven dialogue system, i.e., a system that is meant to be used to perform a specific task, such as booking a ticket, or in the form a non-goal-driven system, for which no such task is given.

2.1 Conversational modelling and dialogue systems

One implementation method for the task of generating dialogue is to use actual lines (possibly slightly modified) from an existing dialogue corpus. This approach was, for instance, applied by Banchs and Li [2012]. They constructed a vector space model-vector from the previous lines in the dialogue, i.e., lines either automatically generated or provided by the human dialogue participant, and measured its distance to vectors constructed in the same fashion from the dialogues in the dialogue corpus. The corpus dialogue which had the closest vector representation was then retrieved, and the dialogue line from the corpus, which was given in response to the ones retrieved, was returned as the next line in the dialogue.

Another solution is to generate new sentences, that do not necessarily have to have been present in the corpus used for training. For this task, neural network techniques are typically applied [Vinyals and Le, 2015; Li *et al.*, 2016; Serban *et al.*, 2016]. For instance, the seq2seq architecture (perhaps best known for its ability to carry out machine translation [Sutskever *et al.*, 2014; Luong *et al.*, 2017]), has been applied for conversational modelling/dialogue generation.

The second approach is intuitively more appealing, since it gives more flexibility to what kinds of lines that can be gen-

erated. Previous studies have, however, shown examples of the generative approach resulting in utterances that are fairly general, as well as examples of that the same utterances are often repeated. That is, the content that is most commonly occurring in the training corpus is that which is most typically being generated, and the potential for flexibility does not automatically lead to a larger creativity. Instead, dialogue lines that are generated mainly on the basis of what is very representative to the corpus might thus be boring in the context of improvisational theatre (and possibly also in most other applications of non-goal-driven dialogue systems). It has been possible to solve the problem of repeated lines, through the application of reinforcement learning that rewards diversity, but the examples provided in the paper describing this approach still include dialogue lines that are rather generic [Li *et al.*, 2016].

In addition, we suspect that the generative approach might require larger dialogue corpora to give usable results, despite that out-of-domain resources, such as large external monologue corpora to initialise word embeddings, have been shown useful [Serban *et al.*, 2016]. Li et al., for instance, used the OpenSubtitles parallel corpus, which consists of around 80 million source-target pairs, for their generative approach.

Since it is relevant to be able to provide automatically generated improvisational dialogues also for languages for which there does not exist a large dialogue corpus and possibly not even a large out-of-domain corpus, or for sub-genres within a language (e.g., improvised Shakespeare [The Improvised Shakespeare Company, 2018] or Strindberg [Strindbergs intima teater, 2012]), it is also important to explore the performance of methods that are less resource demanding. Therefore, along with exploring generative approaches, it might also be relevant to compare these (for different in-domain or out-of-domain training data sizes) to methods that create dialogues through the use of existing dialogue lines.

2.2 Artificial improvisational theatre

The use of artificial intelligence as a part of improvisational theatre has recently been explored by Mathewson and Mirowski [2017]. Their work included the creation of a dialogue system that allowed a human improvisation actor to communicate with a robot that produced lines in response to lines uttered by the human actor. Two versions of the robot dialogue were constructed, one version that selected existing lines in the training corpus, and one version that relied on text generation techniques.

The ambitious approach by Mathewson and Mirowski thus included the use of speech recognition and a text-to-speech system, which functioned in real-time in front of an audience. We believe that this set-up is an appropriate goal for artificial intelligence-powered improvisational theatre, in particular their choice of including a human actor as one of the participants in the dialogue. We suspect, albeit without being able to provide any substantial basis for this suspicion, that watching a human produce lines in real time is one of the main fascinations of improvisational theatre, and that many audience members would quickly lose interest in a play if they were aware of that it only included artificial actors and artificially generated dialogue.

We do, however, not consider this ambitious approach to be appropriate for the goal of objectively evaluating, and thereby in the long run improving, the generation of improvised dialogue. The main reason for this is that the competence of the human actor impacts the quality of the resulting dialogue, since skilful improvisers have a larger ability to fit strange utterances from a co-actor into an improvised scene. There is, for instance, an improvisational theatre game [improwiki, 2018b], where the actors are given a set of pre-written, out-of-context lines, which they are to incorporate in a natural way into the scene. A human actor in an improvisational theatre dialogue is thus very different from a human interacting with a standard, task-oriented dialogue system. In addition, the quality of the text-to-speech system and the speech recognition might influence the audience's perception of the dialogue, and thereby their evaluation of the quality of the dialogue content.

3 Evaluation procedure suggested

Given the problems of including a human actor in a more structured evaluation, we suggest the following procedure for evaluating automatically generated improvisational theatre, in which the task is narrowed down to the generation of dialogue and in which the dialogue is initialised in a manner which increases the possibilities to carry out a reliable evaluation.

3.1 Interaction between two bots

A more reliable evaluation method needs to remove the human influence, and the easiest approach for achieving that would be to replace the human actor with another improvisation bot, i.e., the set-up would be two improvisation bots talking to each other. However, since the end goal is to construct a bot that is able to act against a human actor, the functionality of the bots should not be allowed to be dependent on any one of the bots having full knowledge of the other bot. Instead, the shared knowledge between the two bots should aim to approximate the shared knowledge between two human improvisational actors.

To approximate that level of shared knowledge, we suggest that the two bots that are to be evaluated should not be allowed to be trained on the same training data. The data could be taken from the same text genre, but is should not be the exact same data. That is, in the same manner as two humans that learn the same language are exposed to text from the same genre, i.e., the very wide genre of utterances from many different registers in the language, but are not exposed to the exact same utterances.

3.2 Starting the improvised scene

Improvisational theatre is often carried out with the use of a set of constraints, typically in the form of an input that the actors can use as a starting point for their scene. For instance, the audience could provide an input in the form of a suggestion for a location at which the scene is to take place. Another example is input in form of body postures that the actors use as the starting point for a scene [Johnstone, 1999, pp. 186–187].

The evaluation method we suggest is to use the two initial dialogue lines from human written dialogues as input for the scene. This is a form of input that can be easily automated on a larger scale (as opposed to using non-textual input such as body postures). In addition, the two initial lines provide background data that the dialogue bots can use for generating new lines, which simplifies the task somewhat.

Most importantly, however, using the first two lines of human-written dialogue as input, will result in that the artificially generated dialogue has a comparable human-written dialogue against which it can be evaluated. To make them as comparable as possible, the improvisation bots could be instructed to generate approximately the same number of dialogue lines as the number of lines included in the human dialogue.

3.3 Evaluating the scene from the perspective of its likelihood of having been produced by humans

With the use of this set-up for dialogue generation, for which there will be comparable human-written and automatically generated texts available, the evaluation can be carried out as follows:

The two initial dialogue lines are randomly sampled from a set of (preferably short) human-written dialogues, and one or several pairs of bot systems use these two initial lines to produce a generated dialogue.

A human evaluator is then presented with a set of short dialogue texts, of which some (e.g., half of them) have been selected from the human-written dialogues from which the two initial starter-lines were sampled, and some from the automatically generated dialogues. The task of the human evaluator is then to, for each text, decide whether the dialogue has been generated by a machine or produced by humans. The same human evaluator should not be presented with a human-written dialogue and an automatically generated one that begins with the same two initial lines. With this restriction, the situation that the evaluator carries out a direct comparison between the two texts is avoided. An evaluation through comparison would be a less realistic task, since the final aim is to produce a dialogue that could pass as human-produced, not a dialogue that is *more* human-like than a text that has actually been produced by a human. Employing at least three human evaluators, would be a prerequisite for all automatically generated texts being shown to a human, and that enough texts are annotated to allow for inter-annotator agreement calculations.

Naturally, the set of dialogues from which the two initial dialogue lines are sampled to use as evaluation data, can not be allowed to be included in the data sets used for training the improv-bots.

3.4 Evaluating the scene from other perspectives

There are, of course, other aspects than the resemblance to a human-produced dialogue that the dialogues generated should be evaluated for. Two parameters, mentioned in the background, are the level of diversity among the lines generated and how general the lines produced are. Repetitive and generic dialogue lines are both examples of phenomena that might produce a boring scene, and these two parameters might therefore be combined into a metric in the form of the entertainment value of the dialogues. The evaluator should, therefore, when estimating whether a dialogue has been produced by a human, also assess how entertaining the dialogue is. This is likely to be a more subjective measure. However, given a hypothetical situation in which the artificially generated dialogue often is perceived as being generated by a human, but these dialogues are consistently being given a lower entertainment value score than the human-written dialogues, then this would give an indication of that there is something important missing in the dialogues generated. The easiest solution is, probably, to use a binary score, e.g., to let the evaluator determine whether the dialogue was boring or not.

There are also other types of measures that could be applied for evaluating generated dialogues, e.g., measures that are related to techniques taught within improvisational theatre. An actor should, for instance, aim to be collaborative, e.g., give offers to and accept offers from the co-actors [Johnstone, 1987, pp. 94–108]. To help the audience follow a scene, what roles the actors play, what their relationship is, where the scene is played and what the objectives of the characters are, should also be established early on in a scene [improwiki, 2018a]. It would be a very interesting task to construct an improvisational theatre bot that could achieve such improv-theatre tasks. With these more specific tasks, however, the system is perhaps no longer a non-goal-driven dialogue system, but starts to resemble a goal-driven system. Creating such a system is thus a separate task, for which a separate framework for evaluation should be developed.

4 Implementation

In the long run, we aim to implement and evaluate a resource-intensive method as well, e.g., a method that uses seq2seq to generate new text. However, to illustrate the evaluation method, we here implemented a dialogue creation strategy built on selecting the most appropriate line from a dialogue corpus. This method uses i) a moderate-size dialogue corpus, and ii) a distributional semantics space that is constructed from a very large out-of-domain corpus. We apply a dialogue generation method that is built on several different sub-ideas, which we hope might serve as inspiration for future work, but an evaluation of the contribution of each idea is not within the scope of this paper.

As corpus, we used the Cornell movie-dialogues corpus [Danescu-Niculescu-Mizil and Lee, 2011], and as distributional semantics space we use the word2vec space that has been pre-trained on a very large corpus of Google News and which has been made available by Mikolov et al. [2013; 2013].

Due to the spontaneous and collaborative nature of improvisational theatre, we believe that each dialogue line in this genre in average is likely to be shorter than lines in scripted theatre. We, therefore, extracted a subset of dialogue line triplets from the Cornell movie-dialogues corpus, where each of the lines had to conform to the following set of length criteria: A line was allowed to contain a maximum of two sentences, and in case it contained two sentences, the first of

these two sentences was allowed to contain a maximum of two tokens. The last sentence (that is, the only sentence for one-sentence lines and the second for two-sentence lines) was allowed to contain a maximum of twelve tokens. Sentence splitting and tokenisation was carried out with NLTK [Bird, 2002].

In the Cornell movie-dialogues corpus, there were only 262 dialogues that contained at least six dialogue lines and for which all of the lines conformed to the length criteria we had established for the experiment. These 262 dialogues were, therefore, saved to use as the set of evaluation data, i.e., data which could be used in the evaluation of the automatically generated dialogues. Line triplets from the rest of the corpus were divided into two groups, one group to use as training data for *Actor A* and another group to use as training data for *Actor B*. We divided the triplets film-wise, so that all triplets from the same film were assigned either as training data to *Actor A* or to *Actor B*. In addition, 100 of the dialogues were not added to the training data set, but were used for an informal evaluation during the development, i.e., used as the two first input lines to run the dialogue generation during development. A total of 10,322 line triplets were used to train the functionality for *Actor A* and a total of 10,884 line triplets for the functionality of *Actor B*.

A context in the form of the line most recently uttered in the dialogue and the line before that was used as input data for predicting the next line in the dialogue. The first two lines of each training data triplet were used to represent these two most recent lines, and the third line to represent the line to be predicted. The core of the method for prediction was thus to retrieve the training data triplet for which the two first lines were most similar to the two most recent lines in the generated dialogue, and to use the third line in the triplet as the next line in the generated dialogue. Similarity of dialogue line pairs was determined through converting the two lines into a semantic vector representation, and using the Euclidean distance between the vectors as the similarity measure.

The vector representation for the previous, and the most recently uttered line in the generated dialogue (as well as for the first and second lines in the training data triplets), were constructed as follows: For the previous line, the average of the word2vec vectors representing the tokens in the line were used as the line representation. Tokens present in a standard English stop word list were removed before creating the average vector. For the most recently uttered line, the same representation was used, except that stop words were retained. We believe that also words that are normally considered as stop words are important when interpreting the exact content of the most recently uttered dialogue line, while they might be less important for the content of an earlier line which we included to provide a topical context.

In addition to the averaged vectors, we used the word2vec representation of the three first tokens in the most recently uttered line, as well as the three last tokens in the line, as we believe that these might be more important than the other words for capturing the surface form of the conversation. All of these six vector representations were then concatenated into one long vector. The averaged vectors were slightly downweighted, to give more importance to the vector representations for the three initial and ending tokens of the most recent line (the weights were determined by inspecting the output of the algorithm on the development data). Vector elements were also added to indicate whether a line contained any of the question words *who*, *where*, *when*, *why*, *what*, *which*, *how* or a question mark.

When there were several dialogue line pairs in the training data that matched the lines in the generated dialogues equally well (allowing for a maximum Euclidean distance difference of 0.08 between different candidates), and which resulted in many candidates for the next line, we applied an unsupervised outlier detection to this set of candidates, using scikit-learn's OneClassSVM [Pedregosa *et al.*, 2011]. The set of outliers were then removed from the candidate list.

For the number of candidates that were still present in the candidate list after outliers had been removed, we tried to incorporate the co-operative spirit of improvisational theatre for selecting which of them to use. This was accomplished by selecting the candidate line, for which, when this line (together with its preceding line) was submitted as input the algorithm, the closest neighbour was found. The motivation for this was that when a line was selected to which the co-actor would be more likely to find a good answer, the dialogue would run more smoothly, i.e., just as in real improvisational theatre.

We also applied two simple rules to improve the dialogues, i) to avoid to end a dialogue with a line ending with a question mark, ii) and to avoid repeating a line in the dialogue. These rules were, however, not strictly enforced, and when there were no other candidates of approximately the same quality as a line ending with a question mark or as a repeated line, these were still used.

Word2vec vectors were accessed through the Gensim library [Řehůřek and Sojka, 2010]. The search for dialogue line pairs in the training data, i.e., the dialogue line pairs that were closest to the data given when constructing new dialogues, was sped up by training a scikit-learn NearestNeighbors classifier [Pedregosa *et al.*, 2011].

5 Example output

In Table 1, we present 6 generated dialogues, which were randomly sampled from the set of 262 dialogues that had been set aside as evaluation data. The first two lines are given from the corpus dialogue, and the left-hand column presents the generated version while the right-hand column presents the human-written corpus version. The last two examples show the output of our algorithm and the output presented by Li et al. [2016]. Similarly as when generating lines starting from human-written dialogue, we provided the first two lines in the dialogues published by Li et al. as input to our system.

Our suggested formal evaluation of these dialogues would thus be to present half of the dialogues in Table 1 to *Evaluator 1* and the other half to *Evaluator 2*, who are to determine i) whether the dialogue is produced by a human or not, and ii) whether the dialogue is boring. When informally evaluating these dialogues, we would say that most dialogues in the right-hand column would pass as human made, except the strange dialogue 2, while hardly any of the dialogues in the left-hand column would be classified as produced by humans.

Table 1: The automatically generated dialogues compared to the human-written dialogues, and (for the two last examples), compared to the output of previously published generated dialogue examples. The same human evaluator would either be shown the text in the left-hand column or the text in the right-hand column, and determine i) whether it has been produced by a human, and ii) whether it is boring.

	Computer-generated	Human-written
1	A: I was first to respond. B: What were you doing out here? A: My debrief – B: Change? A: With a vengeance, I hated them. B: She was the Queen of the Netherlands.	A: I was first to respond. B: What were you doing out here? A: Drive by patrol. B: How is she? A: She's tough. B: Have to be. The shit she's gone through.
2	A: I NEED SOME B: YOUNG BLOOD A: On your sleeve. B: Well, I'm sure she kept her word and left you satisfied. A: No. No you're not. B: Yes, Mr. Hillyer, I am. A: Guess we better get going, huh? B: Just a little something' somethin'.	A: I NEED SOME B: YOUNG BLOOD A: I NEED SOME B: YOUNG BLOOD A: I NEED IT NOW I NEED SOME B: YOUNG BLOOD A: I NEED SOME B: YOUNG BLOOD
3	A: I'd better go in. B: I only live up there and across the roof. A: Did you find something? B: Who? A: If he's not out, why are they here? B: I have no idea. A: Why not?	A: I'd better go in. B: I only live up there and across the roof. A: Thanks anyway. B: Listen, Edie, am I going to see you again? A: What for? B: I don't know. A: I really don't know.
4	A: This is ridiculous. B: Government cutbacks. A: Office work? B: You bet it did. A: You drive away business roarin' in like that. B: Bravo!	A: This is ridiculous. B: Government cutbacks. A: Why can't we do it through the mail? B: We missed the deadline. A: Can't we do it on the phone? B: I don't think so.
5	A: Look at my ring. B: That's a fine-looking ring. A: Busy tomorrow? B: Yes. Things are pretty hectic right now, but thanks. A: I know! B: I'm gonna have to ask you to leave.	A: Look at my ring. B: That's a fine-looking ring. A: A lady threw it to me. B: Most surely that was a nice lady to give a ring to a little girl. A: It's a pretty ring. B: I wouldn't be surprised if it were a true wishing ring.
6	A: You all right? B: Yes, thank you. A: Hello, Helen. B: I'm afraid I spoke out of turn last night. A: From whom? B: The main guy, Dieter– A: I don't have time. B: I can see why.	A: You all right? B: Yes, thank you. A: What was that all about? B: He wants me to perform for him. A: Perform? B: You know. A: You on the game? B: God no. I'm a hairdresser.
	Computer-generated here	**Computer-generated in previous studies**
7	A: How old are you? B: I'm 16. Why are you asking? A: Why are you smiling? B: To tell you that we're interested in the property. A: Oh, Dmitri.I wish that I could. B: He didn't say we need guns.	A: How old are you? B: I'm 16. Why are you asking? A: I thought you were 12. B: What made you think so? A: I don't know what you are talking about. B: You don't know what you are saying.
8	A: Where are you going? B: I'm going to the police station. A: I'll help you. B: I'll scream. A: I mean, we get paid tonight, don't we? B: What the fuck is that supposed – A: What?	A: Where are you going? B: I'm going to the police station. A: I'll come with you. B: No, no, no, no, you're not going anywhere. A: Why? B: I need you to stay here. A: I don't know what you are talking about.

None of the dialogues would, however, be classified as boring, except maybe the first of the two dialogues provided by Li et al. [2016], as it starts to generate very generic lines towards the end of the dialogue.

6 Conclusion and outlook

The generated dialogues presented here portray a collection of somewhat strange exchanges, and would not be useful in the context of simulating a real conversation. They might, however, function as absurd dialogues that, for instance, could be used as improvised scene starters. We believe, however, that the more structured form of evaluating a non-goal-driven dialogue system that we present and exemplify could be generally useful. The evaluation structure might be possible to apply in the setting of a shared task, in which the participants not only produce dialogues of this type, but also participate in the evaluation by classifying the dialogues produced by other participating groups.

The next step is to implement a more resource-intensive method, e.g., a method built on seq2seq or some other neural network-based technique. We also intend to extend our initial attempts of achieving dialogue generation with the help of a moderately sized dialogue corpus. We have, for instance, not yet attempted any post-processing of the selected lines to make them fit better into the dialogue, e.g., to make the pronoun gender and number agree between the lines, or to match the use of helper verbs.

Although the ultimate goal would be to achieve an improvbot that could act seamlessly with a human actor, it would also be interesting to explore the suspicion we introduced in the background, i.e., that an audience would quickly lose interest in a play if they were aware of that it consisted solely of artificially generated dialogue. For instance, if two puppets were given two starting lines by the audience, and from these starting lines played a scene with automatically generated human-like dialogues, would the audience still find it interesting?

Acknowledgements

We would like to thank Jonas Sjöbergh, as well as the anonymous reviewers, for valuable input to the content of this paper.

References

[Banchs and Li, 2012] Rafael E. Banchs and Haizhou Li. Iris: a chat-oriented dialogue system based on the vector space model. In *Proceedings of the Association for Computational Linguistics, System Demonstrations*, 2012.

[Bird, 2002] Steven Bird. Nltk: The natural language toolkit. In *Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[Danescu-Niculescu-Mizil and Lee, 2011] Cristian Danescu-Niculescu-Mizil and Lillian Lee. Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011*, 2011.

[improwiki, 2018a] improwiki. crow. https://improwiki.com/en/wiki/improv/crow, 2018.

[improwiki, 2018b] improwiki. Drop a line. https://improwiki.com/en/wiki/improv/drop_a_line, 2018.

[Johnstone, 1987] Keith Johnstone. *Impro : improvisation and the theatre*. Routledge, New York, 1987.

[Johnstone, 1999] Keith. Johnstone. *Impro for storytellers : theatresports and the art of making things happen*. Faber, London, [new ed.] edition, 1999.

[Li *et al.*, 2016] Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. Deep reinforcement learning for dialogue generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Austin, Texas, November 2016. Association for Computational Linguistics.

[Luong *et al.*, 2017] Minh-Thang Luong, Eugene Brevdo, and Rui Zhao. Neural machine translation (seq2seq) tutorial. https://github.com/tensorflow/nmt, 2017.

[Mathewson and Mirowski, 2017] Kory W. Mathewson and Piotr Mirowski. Improvised theatre alongside artificial intelligences. In *In proceedings of AAAI Conference on Artificial Intelligence*. Association for the Advancement of Artificial Intelligence, 2017.

[Mikolov *et al.*, 2013] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.

[Mikolov, 2013] Tomas Mikolov. https://code.google.com/archive/p/word2vec/, word2vec on Google code 2013.

[Pedregosa *et al.*, 2011] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[Řehůřek and Sojka, 2010] Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Paris, France, May 2010. European Language Resources Association (ELRA).

[Serban *et al.*, 2016] Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. Building end-to-end dialogue systems using generative hierarchical neural network models. Proceedings of AAAI, 2016.

[Strindbergs intima teater, 2012] Strindbergs intima teater. http://strindbergsintimateater.se/festival-i-maj-2012/, 2012.

[Sutskever *et al.*, 2014] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[The Improvised Shakespeare Company, 2018] The Improvised Shakespeare Company. http://improvisedshakespeare.com, 2018.

[Vinyals and Le, 2015] Oriol Vinyals and Quoc V. Le. A neural conversational model. *CoRR*, abs/1506.05869, 2015.

[Wikipedia contributors, 2018] Wikipedia contributors. Improvisational theatre — Wikipedia, the free encyclopedia, 2018. [Online; accessed 27-June-2018].