Center for Cyber Security and
International Relations Studies

Fake profiles and BOTs for a fake news factory

The idea of using a fake profile to carry out an incorrect political propaganda is becoming more and more concrete and dangerous, as shown by the recent events that have interfered the last Italian election campaign in anticipation of the voting of March 4th, 2018. In this context, the use of BOTs that autonomously supply for the generation and publication of false information provides a worrying scenario for the future, because they constitute a real fake news factory, whose lies are increasingly difficult to detect.

This paper will show how to recognize a completely automatic proof of concept of BOT within Twitter, describing the necessary requirements to equip it with an artificial intelligence engine, and then illustrating a brief investigation of the potential offered by a such tool.

Introduction

The Turing test allows you to determine whether or not a machine is able to think [1]. In this case, Alan Turing[1] defines an "intelligent machine" as an automatic device capable of concatenating ideas with a complete meaning and expressing them with clarity and naturalness to a human being. Fundamentally, the "robot" should be able to produce expressions that are not meaningless, semantically and syntactically correct. In this case we can actually talk about artificial intelligence.

In the following experiment a BOT was implemented, equipped with a rudimentary artificial intelligence engine based on Markov chains, to create a sort of Turing test distributed on Twitter. The aim of the software developed is to create a fictitious profile within the target social network, which is able to publish fully automatic political expressions of meaning, in order to obtain followers, or political supporters, in view of the upcoming Italian election campaign. In order to avoid generating any inadequate influences on the people, the test was limited to only one month of activity, specifically from 23 September to 26 October 2017. We will see later in the text that this experiment has been carried out successfully, obtaining a certain number of followers, among whom are included even three important politicians. We could therefore say that our BOT was actually able to pass the Turing test (as

[1] Alan Turing was an English mathematician that actually is widely considered to be the father of the theoretical computer science and artificial intelligence.

well as most other BOTs in circulation) and that could be really used to carry out political campaigns, advertising, spam and virtual humint on a large scale.

BOTs as a new tool for unconventional political propaganda

BOT is the abbreviation of Web Robot and it is a software that accesses network services, such as websites, chats and online video games, using exactly the same channels or the same tools used by human beings. The use of these programs allows a massive series of actions to be carried out on a given platform in an almost instantaneous manner, otherwise impossible to carry out by exploiting human labor, or at least within a reasonable time frame.

Social networks are full of BOTs. Some of them communicate with other users of Internet-based services via instant messaging, Internet Relay Chat (IRC), or another web interface. These BOTs allow IRC users to ask questions in plain text and then formulate a proper response [2].

Someone has already thought of using BOTs as a tool for political propaganda in Italy, although episodes of this kind had already occurred in the past even abroad. As confirmed by Intelligence sources, five Twitter accounts have carried out a campaign of disinformation and propaganda on Italian politics during the last months before the elections of March 4^{th}, 2018 [3]. In that article you can read: "the accounts taken in exam have characteristics that do not fall within the activities of normal users of social media". The reason is they are BOTs. One of these BOTs in early 2017 held an average of over 125 daily tweets with consequent results in terms of influence on its followers.

At the moment journalistic sources are not able to extrapolate more detailed information on the matter, but the most important considerations to be made (at least in this context) are the following: is a BOT really able to carry out a political campaign in a fully automatic way? Are the tweeted phrases so credible that they can generate consensus among people? As we will see later in the text, this rudimentary experiment shows that the answer to be given to both questions is affirmative.

Theoretical Description

Our first goal was to be able to identify an artificial intelligence engine that could be simple to manage, but effective enough to be able to generate meaningful sentences independently.

After a preliminary analysis phase we have chosen to use Markov chains in order to generate fake tweets and to publish them into the Internet.

Markov Chains

A Markov chain is a stochastic process, but it differs from a general stochastic process[2]: in fact, a Markov chain must be "memory-less" and (the probability of) future actions are not dependent upon the steps that led up to the present state[3]. This is called the Markov property [4].

[2] A stochastic process is a mathematical object used to represent numerical values related to some system that randomly change over time. This concept is widely used in probability theory and related fields.

In probability theory, the most immediate example is represented by a time-homogeneous Markov chain, in which the probability of any state transition is independent of time [5]. Such a process may be visualized with a labeled directed graph, where the sum of the labels of any vertex's outgoing edges is 1. In other words, knowledge of the previous state is sufficient and necessary to determine the probability distribution of the current state. This definition is broader than the one explored above, as it considers also the non-stationary transition probabilities and therefore time-inhomogeneous Markov chains; therefore, as time goes on (steps increase), the probability of moving from one state to another may change.

We are going to describe the transition matrix, a structure that is used to describe a Markov chain process.

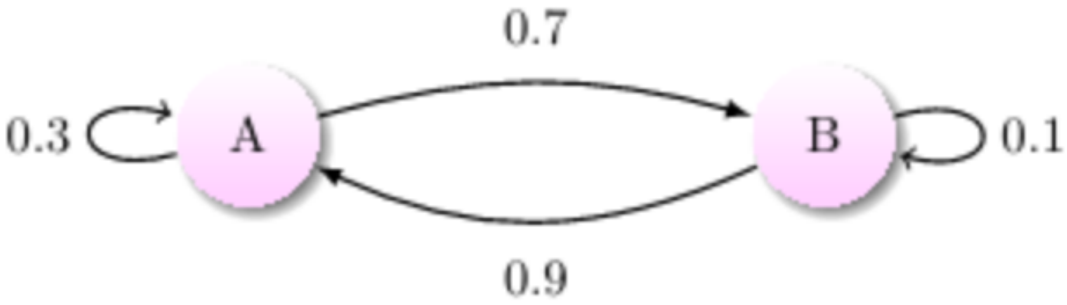
A transition matrix (T) for Markov chain at time is a matrix containing information on the probability of transitioning between states.

The $(i,j)^{th}$ element of the matrix is calculated by $(T)_{i,j} = P(X_{t+1} = j | X_t = i)$.

Then each row of the matrix is a probability vector, and the sum of its entries is 1.

Transition matrices have the property that the product of subsequent ones describes a transition along the time interval spanned by the transition matrices.

This is an example of a two state transition matrix rendered by a labeled directed graph:



Example of a two state Transition Matrix.

The related transition matrix is the following:

$$P = \begin{pmatrix} 0.3 & 0.7 \\ 0.9 & 0.1 \end{pmatrix}$$

Markov chains for text generation

We can use Markov chains to generate artificial texts from an input text corpus. The generated text will be quite similar to that corpus in terms of statistics characterization.

[3] In the theory of discrete dynamical systems, a state is a value wich a stochastic process can take at a certain instant of time.

The algorithm for generating pseudo random text is the following [6]:

- Take N consecutive words from the corpus. We will build a chain of word with a transition matrix.
- Select one initial random state (initial state) and append to the new text (it should be empty at the first step);
- Move randomly to a new state according to the matrix (T);
- Get the text from the state and append to the text that will be generated;
- Repeat the previous step until you reach the desired length of the generated text.

The N parameter is called memory: if you increase this value the text will be more realistic but more similar to the corpus, if you decrease it the text will be more original but more random and less sensible.

For our implementation, due to the limitation of 140 characters for a tweet, we choose $N = 2$. This means that consecutive pairs of words are taken from the corpus of text before being chained by the BOT.

Technical Description

The BOT was implemented in Python[4], using the libraries Tweepy (for Twitter interaction via API) [7] and Markovify (for Markov chain implementation) [8].

We manually select a group of user (around 100) linked to a populist Italian party and we collect the whole tweet corpus storing it in a SQLite database[5] (around 4 milions of tweet).

We then clean up the corpus, removing emoji, links, hashtags and non italian tweets. Furthermore we remove also the non original tweets, filtering out all the retweets. This will create an original tweet corpus (around 1 milions words), making easy to generate a transition matrix for text generation.

The whole dataset is updated and cleaned every 30 minutes. The BOT downloads automatically new tweets from the pool of users followed and then it adds them to the corpus of text.

For creating a more trustable tweet, we decide to develop many models for different timeframe and weight[6], in order to assign more importance to the latest tweets. After some trial we use this scheme:

- Tweet newer than 1 hour: weight 50;

[4] Python is a very popular programming language.

[5] A SQLite datebase is a self-contained database engine based on Structured Query Language (SQL).

[6] A weight is a mathematical object used to influence the result of an average or of a sum. The idea is that elements with more weight will be more influent in the calculation of the result.

- Tweet of the last 4 hour: weight 30;
- Tweet of last day: weight 15;
- Tweet of last week: weight 5;
- Tweet of last month: weight 1;

In order to have a better and realistic text, we then filter out tweet that have entropy[7] less than 4.8 and more than 5.3. These values was selected empirically, after observing many tweet generation. To increase tweet visibility, we randomly add an hashtag from our following in the last two hours.

To simulate a real user, we also like randomly some tweets and make some retweets [9].

Results analysis

After running the BOT for around a month (23 September - 26 October) and posting 330 tweets we have obtained the following result:

- 187 follower;
  - among our followers we have 1 Italian Parliament member, 1 European Parliament member, 1 member of Rome's city council ;
- 15 likes;
- 93 retweets;

In some way, the simple fact of having obtained a number of consents in a short time by real users shows that the generated tweets are credible. Therefore it can be stated that our BOT is potentially able to pass the Turing test successfully.

Other BOT identification

After a couple of weeks it became quite easy to identify some other BOTs. The main criteria to detect them was the following:

- BOT that only retweet/like. We found that some BOTs are not producing any original tweets, but they just retweet or like messages from other users. The majority of these BOT are clearly showing the name of the party or the party logo. We can then assume that they are "institutional" BOT, driven by some organizations linked to the party.
- BOT that generate automatic text. It is very easy to spot some BOTs that generate text in automatic ways. The main symptoms of this behavior are: incorrect grammar, no sense text, tweet hard to read and understand. If a BOT is equipped with a particularly efficient artificial intelligence engine, it is necessary to use a statistical evaluation or a check of the publication of tweets frequency in order to be able to unmask it.

[7] In statistical theory entropy is a quantity that is interpreted as a measure of the disorder present in a system.

Conclusion

Our BOT uses a rather simple artificial intelligence engine, yet it manages to interact adequately and efficiently. To improve it in future developments, the following changes could be considered.

Better text generation

For improving the text readability and hide the automatic text generation, we could filter out Markov chain generated text with some readability and statistical tests. Implementing them will require advanced knowledge of statistical linguistic and an advanced analysis of the corpus.

A more advanced method for generating trustable text will be using LSTM (Long Short Term Memory)[8] instead that Markov chain.

Anyway, LSTM require lot more computational power and a big complexity would be therefore required.

Multiple BOTs coordination

To improve the effect of propaganda on the social media we can create a big cluster of BOTs that posts together in many social networks (Twitter, Instagram, Facebook). This would amplify the message that we want to promote in order to involve a bigger community.

References

[1]. "The turing test." [Online]. Available: https://plato.stanford.edu/entries/turing-test

[2]. K. Graves, CEH Certified Ethical Hacker Study Guide, ser. Serious skills. Wiley, 2010. [Online]. Available: https://books.google.it/books?id=LY1OEbcl1JcC

[3]. "Così la propaganda social filorussa prova a influenzare il voto italiano." [Online]. Available: http://www.ilsecoloxix.it/p/mondo/2018/02/17/ACENIIMB-filorussa_influenzare_propaganda.shtml

[4]. "Markov chains." [Online]. Available: https://www.stat.auckland.ac.nz/~fewster/325/notes/ch8.pdf

[5]. C. M. Grinstead and J. L. Snell, Introduction to Probability. American Mathematical Society, 2012. [Online]. Available: http://www.dartmouth.edu/~chance/teaching_aids/books_articles/probability_book/amsbook.mac.pdf

[6]. D. Beresneva, "Computer-generated text detection using machine learning: A systematic review," in Natural Language Processing and Information Systems, E. Métais, F. Meziane, M. Saraee, V. Sugumaran, and S. Vadera, Eds. Cham: Springer International Publishing, 2016, pp. 421–426.

[8] A LSTM is an active component of a neural network and it makes use of "memory" unlike Markov chains.

[7]. “Tweepy: Twitter for python!” [Online]. Available: https://github.com/tweepy/tweepy

[8]. “Markovify.” [Online]. Available: https://github.com/jsvine/markovify

[9]. “Poetry in python.” [Online]. Available: http://il.pycon.org/2016/static/sessions/omer-nevo.pdf

UNIVERSITÀ
DEGLI STUDI
FIRENZE

DSPS DIPARTIMENTO DI SCIENZE POLITICHE E SOCIALI
DISEI DIPARTIMENTO DI SCIENZE PER L'ECONOMIA E L'IMPRESA

Centro Interdipartimentale di
Studi Strategici, Internazionali e Imprenditoriali - CSSII

Tutti gli scritti pubblicati dal CSSII sono sotto la responsabilità esclusiva dei singoli autori