6.864 project report

Marina Munkhoeva

December 2015

1 Abstract

This project aimed to perform binary classification on fanfiction pieces by means of LSTM network. Support vector machines has been is chosen to be baseline. Trained vector representations for the main characters of the book series show interesting patterns for canonical and alternative universe(AU) fan fiction. [1]

2 Introduction

Fan fiction is usually stories written by fans of some fictional universe involving its settings and characters. The problem of this project is to try discriminate between canonical (compatible with the original storyline) and non canonical (contradicting original work) fan fiction. For example, any fan of Harry Potter universe could discriminate the following excerpt:

"Mum," Harry said. "If you want to win this argument with Dad, look in chapter two of the first book of the Feynman Lectures on Physics."

as belonging to alternative universe. However, it takes more effort to label:

"... he turned on the spot and vanished, leaving nothing behind but the bag Ron had snatched from his hand as he went and some flying chunks of vomit..."

as belonging to the original piece of work. The last example is also a marker of how canonical pieces of text cannot be actually discriminated on the level of sentences with high confidence. The issue is that this classification problem is highly dependent of facts and to solve it we should find a good representation for the facts to be able to learn on them.

Empirically most common and useful indicator of alternative universe fan fiction is a notion of pairings. Pairing X/Y denotes that the story involves character X dating/being married to/engaged to character Y. Since enthusiasts love to explore what could happen between a pair of characters in fan fiction this

[1] https://github.mit.edu/maremun/fanfiction

even produced pairing generators for plot[2]. Thus since most of the canonicity is contained in the relationships between characters it does make sense to look into character representations in fan fiction.

While the problem can be solved on the basis of ngram representation of texts with good accuracy, it is very tempting to apply Long Short Term Memory recurrent neural network to discriminate between 'canon' and AU. LSTM is known to be very good at capturing structure and long term time dependencies. Since fiction is usually quite prolonged text it makes sense to try LSTM. It is also interesting to see whether LSTM can learn facts corresponding to canonical storyline from the text.

3 Data Acquisition and Preprocessing

There are a bunch of fictional universes, for instance *Star Trek, Middle Earth (The Lord of The Rings)*, etc. This project will work with Joanne Rowling's *Harry Potter* series and corresponding fan fiction.

I am using fan fiction from `http://www.harrypotterfanfiction.com/`. The web resource holds overall 80000+ stories with 300000+ chapters. Most of the fan fiction is annotated with short description: author, genres, format, rating, abstract, pairings, etc. Most of the stories fall into two categories in terms of format: **Novel** (consisting of several relatively long chapters, usually 3-4 thousand words) and **Short story** (varies in length, but also around 2k words per chapter). Although I mentioned earlier that canon/AU classification can be resolved through pairings, such thing would not be much of a natural language processing, but still can be a nice baseline (80% accuracy) for the task.

Each piece of fan fiction usually holds several genres (**adventure, drama, general, humor, mystery**, etc). Whenever there is **au** (alternative universe) in the genres annotation the story is considered to be non canonical, otherwise it is canonical. I will also include *Muggle's Guide to Harry Potter*[3]. The guide contains summary of each chapter of the original book series.

I have parsed the mentioned web resource htmls to get the positive and negative samples of fan fiction. I split the corpus into the level of chapters to run SVM. For vector representation learning I picked random subsets of canon and AU fan fiction chapters to learn on. For LSTM training examples are sentences of 30 words.

4 Baseline

SVM on normalized bigrams (occurrence threshold = 10) has been used to mark a base performance for the problem. I have tried two variants for the training

[2]http://darkhp.childofthemorning.com/

[3]http://en.wikibooks.org/wiki/Muggles%27_Guide_to_Harry_Potter/Books/

set: with and without original chapters. Each entry in the training and test sets corresponds to chapter in fiction piece.

Having two options for the training set helps to see if performance on fan fiction test set improves with adding original chapters to the training set. This does help indeed, raising score for fixed training and test sets by 4%. In general, SVM yields around 60-70% accuracy.

5 LSTM

I have tried to train LSTM. My plan was to use pre-trained word embeddings[4]. I started with single LSTM layer followed by mean pooling and logistic regression output layer. After that I wanted to run training GloVe[5] vectors learned on complete fan fiction corpus ($388m$ tokens with $134k$ unique words,occurrence threshold = 5) and see what is the difference.

As mentioned earlier the training examples would be sentences of 30 words. It seems extremely unlikely that feeding a prolonged paragraph or even chapter would produce anything sensical by the model I chose as producing paragraph/document embeddings is one of hot topics in natural language processing[6] and requires more complex approach.

However I encountered technical issues with Keras library and could not resolve it and realised I have little time to produce my own implementation of LSTM on using Theano.

6 Vector representation analysis

Also I have tried running GloVe on the original chapters and have obtained vector representations for words to try analyze relationships between characters.

As mentioned in introduction I aimed to analyze canonical and non canonical difference through character closeness. I found out that two major patterns in fan fiction are well represented with trained character vectors:

1. non canonical fan fiction tends to put characters in new settings apart from his friends and usual environment,

2. canonical is more about future of the characters, while non canonical is focused on time represented in original series.

For example, Harry of canonical universe (see Table 2) as vector is surrounded by his friends, while Harry of alternative (see Table 2) is not. Draco Malfoy is very close to his present family members (Licius and Narcissa) in alternative representation and to his future family members (Astoria and Scorpius) in canonical.

[4] http://code.google.com/p/word2vec/
[5] http://nlp.stanford.edu/projects/glove/
[6] http://aclweb.org/anthology/D15-1280, http://arxiv.org/pdf/1506.01057v2.pdf

harry	hermione	ron	**draco**	ginny	sirius
ron	ginny	ginny	malfoy	hermione	james
again	ron	fred	**lucius**	ron	remus
suddenly	draco	george	turned	red	peter
said	luna	hermione	raised	luna	lily
remus	finally	neville	**blaise**	george	shook
see	fred	harry	moment	lily	sighed
finally	again	surprised	again	finally	rolled
him	surprised	seamus	**narcissa**	tonks	tonks
could	looking	tonks	stood	neville	regulus

Table 1: 9 nearest neighbours for character vectors in alternative universe

harry	hermione	ron	**draco**	ginny	sirius
ron	ginny	harry	malfoy	hermione	james
hermione	ron	hermione	**scorpius**	ron	remus
ginny	harry	george	**astoria**	harry	peter
george	george	ginny	**blaise**	george	lily
bill	luna	charlie	simply	molly	smirked
draco	draco	fred	luna	shook	rolled
charlie	again	bill	once	luna	laughed
said	sighed	teddy	again	sighed	severus
once	molly	again	granger	bill	fred

Table 2: 9 nearest neighbours for character vectors in canonical universe

To be able to study the nature of relationship (friendship/opposition/etc) I visualize character vectors structure under assumption that the vector direction between characters in canonical and alternative universes represent different types of relationship. This is done by reducing dimensionality of character embeddings via t-SNE[7].

7 Conclusion

This project has utilized Support Vector Machines approach to get a good baseline for the classification task. Determining canonical and alternative universe fan fiction is mostly represented by character relationships. This has been studied using character vector representations. We saw closeness of characters in both types of fan fiction. However closeness of characters does not tell you much about the tone of the relationship, so we took a look at the structure of character points in vector space learned on the whole corpus.

[7]https://lvdmaaten.github.io/tsne/