Cs 224 D Final
Crossing Minds - Semantic Cross Recommendation

Alexandre Robicquet
Department of Computer Science
Stanford University
arobicqu@stanford.edu

Abstract

Recommendation engines are common among e-commerce, targeted advertising, social media and content-based websites. Amazon was one of the first websites to use a recommendation system. When the company was essentially an online book store, it began using software to suggest books the user might be interested in, based on data gathered about their previous activity, as well as the activity of other users who made similar choices. Beyond all those points, we tried to create the first platform of smart and evolutive Cross Recommendations, based on the idea that The better I know any of your taste, the more relevant I will be. In other words, a unique platform assembling the state-of-the-art algorithms in deep learning and optimization in order to expand the recommendation system field to an all other level The deep learning architecture is the perfect tool to merge every data into our recommendations. It performs automatic selection of relevant input and discards the useless or mis-informative data. The treatment of structured data such as text or pictures requires state-of-the-art machine learning. In order to include text data, we are are delving into the Natural Language Understanding front. We get loads of natural language data that can be in the form character descriptions, movie synopsis, reviews of places, hotels etc. Then we use state-of-the-art home grown methods to convert these into mathematical entities called tensors. Once we have converted natural language into a coarse mathematical representation, we run a suite of analysis tools that extract the true semantic meaning of the text, which enables us to match it against other items in our dataset thus enabling truest retrievals that are most similar to the user query

1 Introduction

In recent experiments we have seen that word2vec has proved as an efficient method for learning high-quality distributed vector representations that capture a large number of precise syntactic and semantic word relationships. An inherent limitation of word representations is their indifference to word order and their inability to represent idiomatic phrases. The word2vec also seems to be limited in its application, because when we convert large pieces of text into vectors, we either average the word2vec of each word in the sentence, or build a representation by modelling the temporal dependencies of such vectories externally using a Recurrent Neural Network extension. With this motivation, we present several extensions that improve both the quality of the vectors and the training speed. We use an approach for unsupervised learning of a generic, distributed sentence encoder. Using the continuity of text from books, we train an encoderdecoder model that tries to reconstruct the surrounding sentences of an encoded passage. After training our model, we extractand evaluate our vectors with linear models on some very famous taskes like semantic relatedness, etc.

Developing learning algorithms for distributed compositional semantics of words has been a long-standing open problem at the intersection of language understanding and machine learning. In recent

years, several approaches have been developed for learning composition operators that map word vectors to sentence vectors including recursive networks [1], recurrent networks [2], convolutional networks [3, 4] and recursive-convolutional methods [5, 6] among others. The latest version and one we will focus our work on is proposed in such as an approach for unsupervised learning of a generic, distributed sentence encoder. Using the continuity of text from books, they trained an encoder-decoder model that tries to reconstruct the surrounding sentences of an encoded passage. Sentences that share semantic and syntactic properties are thus mapped to similar vector representations.

In this

2 Training And Dataset

To lead our experiments, we gather a unique a very large dataset of informration for each data we would like to recommend our user with: music, tv-shows, movies, books. To this extend, we juggle throught different API and gather the following content introduce in table:

Type	Source	Text	hashtags
Music	Last Fm	59301	17214
Movies	Imdb	103975	30146
Tv-shows	Imdb	7378	4952
Books	Goodreads	508901	268102

Figure 1: Collected Datasets

Moreover, in order to test our implpementation and semantic cross-recommendation system, we scrapped the users like and dislike from a taste-collection website named crossing mind. With a total of 80530 users, the quantitative description of this collection is presented in figure 2.

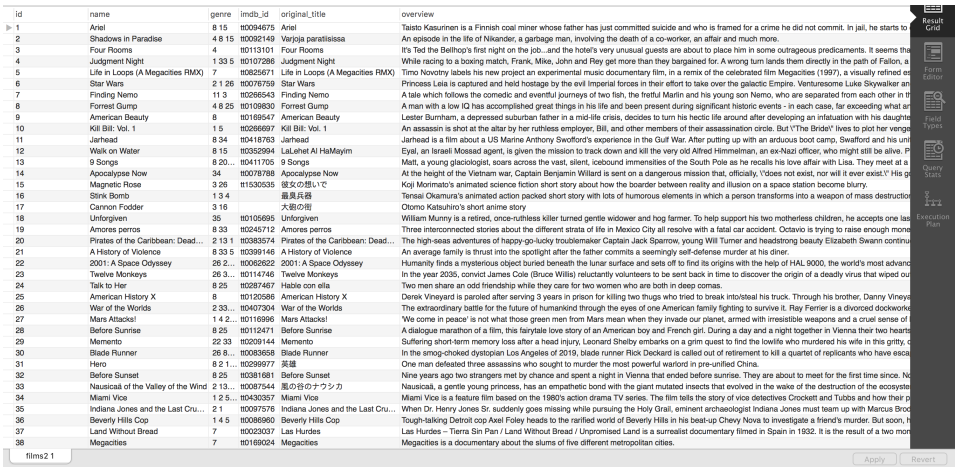
Data	Positives	Negatives	Total
Music	119301	9598	128899
Movies	203975	8786	212761
Tv-shows	17864	464	18328
Books	19884	3089	22973

Figure 2: Tastekid Collection

All those dataset would be available as sql dataset during the summer, containing the synopsis of the movies, the biographie of the artist, the story of the books and the plot of the tv-shows. A figure of our work is presented in figure 3.

Initial Model

Before going into the details of the Neural Network, let's simplify the idea to the extreme and consider a linear model. We are given a data set of likes and dislikes of users on items. Let's n_u and n_i be the numbers of users and items. We can model the data set with a sparse matrix L of size (n_u, n_i) where $L[i, j] = 1$ if the user i likes the item j, or L[i,j]=-1 if the user dislikes the item, and $L[i, j] = 0$ if we don't know. The linear model assumes that we can find for each user and each item a vector in Rd such that the like value is approximately equals to the dot product of the user's vector and the item's vector. The components of those vectors are called features. The features for the user can be viewed as affinity for the item's feature. In this simple model a user likes an item if it has high affinity for the item's highest features. Given L, the learning problem is then to find a matrix U of size (n_u, d) and a matrix V of size (n_i, d) such that $L \sim UVT$. The parameter d has to be cross-validated. The higher d is, the highest we will overfit the data set. Once we have U and V, we can recommend to the user i the item j maximizing $U_i V_j T$. Finding this optimal item in time less than O(n_i) is possible with the Top-K algorithm.



id	name	genre	imdb_id	original_title	overview
1	Ariel	8 15	tt0094675	Ariel	Taisto Kasurinen is a Finnish coal miner whose father has just committed suicide and who is framed for a crime he did not commit. In jail, he starts to
2	Shadows in Paradise	4 8 15	tt0092149	Varjoja paratiisissa	An episode in the life of Nikander, a garbage man, involving the death of a co-worker, an affair and much more.
3	Four Rooms	4	tt0113101	Four Rooms	It's Ted the Bellhop's first night on the job...and the hotel's very unusual guests are about to place him in some outrageous predicaments. It seems tha
4	Judgment Night	1 33 5	tt0107286	Judgment Night	While racing to a boxing match, Frank, Mike, John and Rey get more than they bargained for. A wrong turn lands them directly in the path of Fallon, a
5	Life in Loops (A Megacities RMX)	7	tt0825671	Life in Loops (A Megacities RMX)	Timo Novotny labels his new project an experimental music documentary film, in a remix of the celebrated film Megacities (1997), a visually refined es
6	Star Wars	2 1 26	tt0076759	Star Wars	Princess Leia is captured and held hostage by the evil Imperial forces in their effort to take over the galactic Empire. Venturesome Luke Skywalker an
7	Finding Nemo	11 3	tt0266543	Finding Nemo	A tale which follows the comedic and eventful journeys of two fish, the fretful Marlin and his young son Nemo, who are separated from each other in th
8	Forrest Gump	4 8 25	tt0109830	Forrest Gump	A man with a low IQ has accomplished great things in his life and been present during significant historic events - in each case, far exceeding what an
9	American Beauty	8	tt0169547	American Beauty	Lester Burnham, a depressed suburban father in a mid-life crisis, decides to turn his hectic life around after developing an infatuation with his daughte
10	Kill Bill: Vol. 1	1 5	tt0266697	Kill Bill: Vol. 1	An assassin is shot at the altar by her ruthless employer, Bill, and other members of their assassination circle. But \"The Bride\" lives to plot her venge
11	Jarhead	8 34	tt0418763	Jarhead	Jarhead is a film about a US Marine Anthony Swofford's experience in the Gulf War. After putting up with an arduous boot camp, Swafford and his unit
12	Walk on Water	8 15	tt0352994	LaLehet Al HaMayim	Eyal, an Israeli Mossad agent, is given the mission to track down and kill the very old Alfred Himmelman, an ex-Nazi officer, who might still be alive. P
13	9 Songs	8 20…	tt0411705	9 Songs	Matt, a young glaciologist, soars across the vast, silent, icebound immensities of the South Pole as he recalls his love affair with Lisa. They meet at a
14	Apocalypse Now	34	tt0078788	Apocalypse Now	At the height of the Vietnam war, Captain Benjamin Willard is sent on a dangerous mission that, officially, \"does not exist, nor will it ever exist.\" His g
15	Magnetic Rose	3 26	tt1530535	彼女の想いで	Koji Morimato's animated science fiction short story about how the boarder between reality and illusion on a space station become blurry.
16	Stink Bomb	1 3 4		最臭兵器	Tensai Okamura's animated action packed short story with lots of humorous elements in which a person transforms into a weapon of mass destruction
17	Cannon Fodder	3 16		大砲の街	Otomo Katsuhiro's short anime story
18	Unforgiven	35	tt0105695	Unforgiven	William Munny is a retired, once-ruthless killer turned gentle widower and hog farmer. To help support his two motherless children, he accepts one las
19	Amores perros	8 33	tt0245712	Amores perros	Three interconnected stories about the different strata of life in Mexico City all resolve with a fatal car accident. Octavio is trying to raise enough mone
20	Pirates of the Caribbean: Dead…	2 13 1	tt0383574	Pirates of the Caribbean: Dead…	The high-seas adventures of happy-go-lucky troublemaker Captain Jack Sparrow, young Will Turner and headstrong beauty Elizabeth Swann continu
21	A History of Violence	8 33 5	tt0399146	A History of Violence	An average family is thrust into the spotlight after the father commits a seemingly self-defense murder at his diner.
22	2001: A Space Odyssey	26 2…	tt0062622	2001: A Space Odyssey	Humanity finds a mysterious object buried beneath the lunar surface and sets off to find its origins with the help of HAL 9000, the world's most advanc
23	Twelve Monkeys	26 3…	tt0114746	Twelve Monkeys	In the year 2035, convict James Cole (Bruce Willis) reluctantly volunteers to be sent back in time to discover the origin of a deadly virus that wiped ou
24	Talk to Her	8 25	tt0287467	Hable con ella	Two men share an odd friendship while they care for two women who are both in deep comas.
25	American History X	8	tt0120586	American History X	Derek Vineyard is paroled after serving 3 years in prison for killing two thugs who tried to break into/steal his truck. Through his brother, Danny Vineya
26	War of the Worlds	2 33…	tt0407304	War of the Worlds	The extraordinary battle for the future of humankind through the eyes of one American family fighting to survive it. Ray Ferrier is a divorced dockworke
27	Mars Attacks!	1 4 2…	tt0116996	Mars Attacks!	'We come in peace' is not what those green men from Mars mean when they invade our planet, armed with irresistible weapons and a cruel sense of
28	Before Sunrise	8 25	tt0112471	Before Sunrise	A dialogue marathon of a film, this fairytale love story of an American boy and French girl. During a day and a night together in Vienna their two hearts
29	Memento	22 33	tt0209144	Memento	Suffering short-term memory loss after a head injury, Leonard Shelby embarks on a grim quest to find the lowlife who murdered his wife in this gritty, c
30	Blade Runner	26 8…	tt0083658	Blade Runner	In the smog-choked dystopian Los Angeles of 2019, blade runner Rick Deckard is called out of retirement to kill a quartet of replicants who have esca
31	Hero	8 2 1…	tt0299977	英雄	One man defeated three assassins who sought to murder the most powerful warlord in pre-unified China.
32	Before Sunset	8 25	tt0381681	Before Sunset	Nine years ago two strangers met by chance and spent a night in Vienna that ended before sunrise. They are about to meet for the first time since. No
33	Nausicaä of the Valley of the Wind	2 13…	tt0087544	風の谷のナウシカ	Nausicaä, a gentle young princess, has an empathetic bond with the giant mutated insects that evolved in the wake of the destruction of the ecosyste
34	Miami Vice	1 2 5…	tt0430357	Miami Vice	Miami Vice is a feature film based on the 1980's action drama TV series. The film tells the story of vice detectives Crockett and Tubbs and how their p
35	Indiana Jones and the Last Cru…	2 1	tt0097576	Indiana Jones and the Last Cru…	When Dr. Henry Jones Sr. suddenly goes missing while pursuing the Holy Grail, eminent archaeologist Indiana Jones must team up with Marcus Broc
36	Beverly Hills Cop	1 4 5	tt0086960	Beverly Hills Cop	Tough-talking Detroit cop Axel Foley heads to the rarified world of Beverly Hills in his beat-up Chevy Nova to investigate a friend's murder. But soon, h
37	Land Without Bread	7	tt0023037	Las Hurdes	Las Hurdes – Tierra Sin Pan / Land Without Bread / Unpromised Land is a surrealist documentary filmed in Spain in 1932. It is the result of a two mon
38	Megacities	7	tt0169024	Megacities	Megacities is a documentary about the slums of five different metropolitan cities.

Figure 3: Example of the sql scrapping database

Why a Neural Network ?

The major limitation of the previous model is that we cannot fit any non-linearity between users' features and items' features. Instead of having P[i likes j] kd U[i,k] V[j,k], we may want

$$P[ilikesj] \sum_{kd} \Phi(U[i,k]V[j,k])$$

, where Φ is a non-linear increasing function. The learning of U and V is rendered much more difficult than before, hence the use of neural network learning algorithm. The values for U and V are learnt by stochastic gradient descent and backward propagation of the learning error. Since Φ is monotone, it is still possible to use the Top-K algorithm to recommend the item maximizing the like prediction for a user. Including various data sets

The second motivation for using a neural network is than it becomes handy to learn from various data sets. Indeed if we have several likes data sets such as movies likes and musics likes, we can use shared features between both, so that the NN will learn jointly on movies and musics. What's more, appart from the likes data we may possess some attributes of the items such as tags, actors, kinds, etc. It is possible to include this additional information in the learning process and make the NN utilizing all the data set together to improve the predictions. To do so, create additional m features from these attributes and concatenate them to the items' features vector. The new items' and users' features vector are now of dimension d+m. Note that the m last dimensions of the items' vector are not learnt but fixed according to the attributes.

To this model, we want to introduce and add some semantic aspect or in other term: how can we extract the spirit of a story to improve our cross-recomendation?

3 sent2vec

We use and improve in our approach the skip-thought model introduced in. The skip-thought is an encoder which map words to a sentence vector and a decoder is used to generate the surrounding sentences. Encoder decoder models have gained a lot of traction for neural machine translation. In this setting, an encoder is used to map e.g. an English sentence into a vector. The decoder then conditions on this vector to generate a translation for the source English sentence. A clear example of this approch is given in figure 4

Encoder. Let $w_i^1, ..., w_i^N$ be the words in sentence si where N is the number of words in the sentence. At each time step, the encoder produces a hidden state h_i^t which can be interpreted as the representation of the sequence $w_i^1, ..., w_i^t$. The hidden state h_i^N thus represents the full sentence. To encode a sentence, we iterate the following sequence of equations (dropping the subscript i)

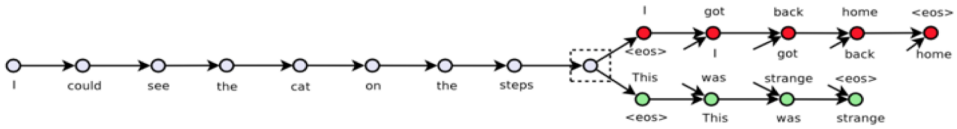


Figure 4: The skip-thoughts model. Given a tuple (s_{i-1}, s_i, s_{i+1}) of contiguous sentences, with s_i the i-th sentence of a book, the sentence s_i is encoded and tries to reconstruct the previous sentence s_{i-1} and next sentence s_{i+1}. In this example, the input is the sentence triplet *I got back home. I could see the cat on the steps. This was strange.* Unattached arrows are connected to the encoder output. Colors indicate which components share parameters. ⟨eos⟩ is the end of sentence token.

$$r^t = \sigma(W_r x^t + U_r h^{t1}) \tag{1}$$
$$z^t = \sigma(W_z x^t + U_z h^{t1}) \tag{2}$$
$$\bar{h}^t = \tan h(W x^t + U(r \odot h^{t1}) \tag{3}$$
$$h^t = (1 - z^t) \odot h^{t-1} + z^t \odot \bar{h}^t \tag{4}$$

where $\bar{h}^t$ is the proposed state update at time t, z^t is the update gate, r_t is the reset gate $\odot$ denotes a component-wise product. Both update gates takes values between zero and one.

Decoder. The decoder is a neural language model which conditions on the encoder output h_i. The computation is similar to that of the encoder except we introduce matrices C_z, C_r and C that are used to bias the update gate, reset gate and hidden state computation by the sentence vector. One decoder is used for the next sentence s_{i+1} while a second decoder is used for the previous sentence s_{i1}. Separate parameters are used for each decoder with the exception of the vocabulary matrix V, which is the weight matrix connecting the decoder's hidden state for computing a distribution over words. In what follows we describe the decoder for the next sentence s_{i+1} although an analogous computation is used for the previous sentence s_{i1}. Let h^t_{i+1} denote the hidden state of the decoder at time t. Decoding involves iterating through the following sequence of equations (dropping the subscript i + 1):

$$r^t = \sigma(W^d_r x^{t-1} + U^d_r h^{t1} + C_r h_i) \tag{5}$$
$$z^t = \sigma(W^d_z x^{t-1} + U^d_r h^{t1} + C_z h_i) \tag{6}$$
$$\bar{h}^t = \tan h(W^d x^{t-1} + U^d(r^t \odot h^{t1}) + C h_i) \tag{7}$$
$$h^t_{i+1} = (1 - z^t) \odot h^{t-1} + z^t \odot \bar{h}^t \tag{8}$$

Given h^t_{i+1}, the probability of word w^t_{i+1} given the previous t 1 words and the encoder vector is

$$P(w^t_{i+1} | w^{<t}_i + 1, h_i) \sim exp(v_{w^t_{i+1}} h^t_{i+1})$$

where $v_{w^t_{i+1}}$ denotes the row of V corresponding to the word of w^t_{i+1}. An analogous computation is performed for the previous sentence s_{i1}.

Objective

Given a tuple (s_{i-1}, s_i, s_{i+1}), the objective optimized is the sum of the log-probabilities for the forward and backward sentences conditioned on the encoder representation:

$$\sum_t \log P(w^t_{i+1} | w^{<t}_{i+1}, \mathbf{h}_i) + \sum_t \log P(w^t_{i-1} | w^{<t}_{i-1}, \mathbf{h}_i)$$

4 Experiments

Our main experiment was the direct application and improvement of the skip-thought algorithm. By doing so, we are going throught the following steps to evaluate the capability of the encoder as a generic feature extractor after training on the BookCorpus dataset introdce in [8]. Our experimentation setup on each task is as follows:

- Using the learned encoder as a feature extractor, extract skip-thought vectors for all sentences.
- If the task involves computing scores between pairs of sentences, compute component-wise features between pairs. This is described in more detail specifically for each experiment.
- Train a linear classifier on top of the extracted features, with no additional fine-tuning or backpropagation through the skip-thoughts model.

For the purposes of this report, the only set of experiments that have been completed as of now, is the trianing of the skip thought inspired vectors themselves. We train a single models on our book corpus. It consists of a unidirectional encoder with 2400 dimensions, which we subsequently refer to as uni-skip. We also plan to train another model which would be a bidirectional model with forward and backward encoders of 1200 dimensions each. This model contains two encoders with different parameters: one encoder is given the sentence in correct order, while the other is given the sentence in reverse. The outputs are then concatenated to form a 2400 dimensional vector. This is referred to as a bi-skip model in the original skip thoughts vector paper

For training, we initialize all recurrent matricies with orthogonal initialization. Non-recurrent weights are initialized from a uniform distribution in [-0.1,0.1]. Mini-batches of size 128 are used and gradients are clipped if the norm of the parameter vector exceeds 10. We used the Adam algorithm for optimization. The model was trained for 4 days on a NVidia Titan X chip and the implementation was done using Theano.

Thus far, we get a cross entropy cost of 0.6317 and we hope to improve it further training the model longer and by also including some optimizations that were covered in the course.

the results of our model are introduced in figure 5:

As we can see through our results, we could be better at recomeneding what someone might like (TPR) but not at all at determining what someone don't llike (TNR)

Conclusion

PLease, go try our beta version and do not hesitate to give us your feedback:

$$https://privatebeta.crossingminds.org/$$

Password:

$$BWFFVDMRFX377HZ8F2ATDUFVIG0REDX7$$

References

[1] Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In EMNLP, 2013.

[2] Sepp Hochreiter and Jrgen Schmidhuber. Long short-term memory. Neural computation, 9(8):1735 1780, 1997.

[3] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. ACL, 2014.

[4] Yoon Kim. Convolutional neural networks for sentence classification. EMNLP, 2014.

[5] Kyunghyun Cho, Bart van Merrinboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. SSST-8, 2014.

[6] Han Zhao, Zhengdong Lu, and Pascal Poupart. Self-adaptive hierarchical sentence model. IJCAI, 2015.

[7] Quoc V Le and Tomas Mikolov. Distributed representations of sentences and documents. ICML, 2014.

15(7):5249-5262.

v0.1

No NLP data. 16 epochs.

v1.1.1

One layer NLP data (256 dimensions) + dropout = 0.1 + no batchNorm

v1.1.2

One layer NLP data (1024) + dropout + no BatchNorm

v1.2.0

Two layer NLP data (512 - 256) + dropout = 0.1 + no batchNorm

v1.2.1

Two layer NLP data (512 - 64) + dropout = 0.1 + batchNorm(only second layer)

v1.3

Three layer NLP data (1024 + 512 + 256) + dropout = 0.1 + no batchNorm

version	avg F-mes	TPR	TNR	1st epoch
v0.1	0.87572733	0.77766746	0.43294895	0.86773863
v1.1.1	0.85862448	0.71708246	0.38235965	
v1.1.2	0.86340604	0.71531951	0.3888991	0.85862448
v1.2.0	0.84529701	0.82490516	0.27257456	0.85214741
v1.3	0.85587623	0.78227022	0.30812605	0.85188662

Figure 4: Results of adding the skip-thought process to our initial model according to different architecture 9TPR true positive rating - NR true negative rating)