Towards Text-Based Recommendations

Damien Poirier
Orange Labs
2 avenue Pierre Marzin
22300 Lannion, FRANCE
damien.poirier@orange-ftgroup.com

Isabelle Tellier
LIFO, Université d'Orléans
Rue Léonard de Vinci
45000 Orléans, FRANCE
isabelle.tellier@univ-orleans.fr

Françoise Fessant
Orange Labs
2 avenue Pierre Marzin
22300 Lannion, FRANCE
francoise.fessant@orange-ftgroup.com

Julien Schluth
GFI Informatique
11 rue Louis Broglie
22300 Lannion, FRANCE
julien.schluth@gmail.com

ABSTRACT
Recommender systems have become, like search engines, a tool that cannot be ignored by a website with a large selection of products, music, news or simply webpages. The performance of this kind of system depends on a large amount of information. At the same time, the amount of information on the Web is continuously growing, especially due to increased User Generated Content since the apparition of Web 2.0. The great majority of this content is composed of unstructured textual data. In this paper, we propose a method allowing for recommendation with only textual data. The method we propose has two steps. First, subjective texts are labelled according to their expressed opinion in order to build a user-item-rating matrix. Second, this matrix is used to establish recommendations thanks to a collaborative filtering technique. We describe the complete processing chain and evaluate it.

Keywords
Opinion classification, Recommender systems, User Generated Content, Collaborative filtering

1. INTRODUCTION
The aim of recommender systems is to help users find items that they would appreciate from large web-based catalogues. Items can be of any type such as movies, music, books, webpages, etc.. To do so, recommender systems use information about past user preferences in order to select items that could be interesting for them. The main challenge to build an efficient recommender system is to collect enough data to "initialize" the recommendation process. The idea developped in this work is that it is possible to collect these data from texts found on the web. As a matter of fact, the web is now a huge reservoir of information that is continuously growing, especially thanks to User Generated Content, appeared with the explosion of the Web 2.0. The great majority of this content is composed of unstructured textual data. To fill the gap between opinion texts and recommendation, we need to refer to the growing field of opinion mining [3]. In this article, we will try to provide ratings inferred from opinion texts to feed a recommender system. To evaluate our approach, we choose the well-studied domain of recommendation about movies.

2. PROCESSING CHAIN
The processing chain we propose can be separated in two main tasks. The first one concerns the analyzis of textual data. This task consists in obtaining a user-item-rating matrix. The choice of texts is then important. Each text has to contain an opinion about an identified item and the author of the text has to be known too. Once a large number of user-item-review triplets are stocked, the opinion classification task can be applied in order to infer a rating for each review, i.e. in order to create user-item-rating triplets. The majority of studies on opinion classification have chosen a two classes classification (like, dislike) or a three classes classification (like, neutral, dislike). The second tasks consists in generating recommendations. After building the user-item-rating matrix, collaborative filtering can be applied and recommendations can be done. For this step, the recommender system builds a similarity matrix of users or items thanks to the user-item-rating matrix. We decided to test the process chain on movies recommendation. It is a very common application domain in the recommender systems field and some benchmarks are available. In order to test the process chain, we took three corpora for our experiments. The first and second one are textual corpora while the third one is a corpus of ratings.

The corpus of ratings, called *Corpus 3*, is used to evaluate quality of recommendations, which is the last test of the experiments. This *Corpus 3* is a well-known corpus in the field of Recommender Systems. It is the set of logs made available by Netflix[1] for the Netflix Prize, a challenge created in order to improve the collaborative filtering algorithms for an online DVD-rental service. It contains ratings put on Netflix by 400,000 users, that is approximately 100,000,000 user-item-rating triples. *Corpus 2* comes from the community website Flixster[2]. Flixter is a community space where a lot of movie fans meet and share their tastes and preferences about cinema. They can create a personnal page where they put, among other things, ratings and reviews about films. This corpus, which is dedicated to the test of the opinion classification task, is composed of approximately 3,330,000 user-item-rate-review quadruples where reviews were writ-

[1]www.netflix.com
[2]www.flixster.com

ten by almost 100,000 users and speak about 10,500 movies. All of these 10,500 movies are also present in *Corpus 3*. *Corpus 1* also comes from Flixster but it has no intersection with *Corpus 2*. It contains approximately 175,000 user-item-rate-review quadruples. It is used during the learning step of the opinion classification .

Preprocessings applied to textual corpora (*Corpus 1* and *Corpus 2*) are few and light. The only treatments done consists in putting every letter in lowercase and deleting words with less than three occurences in *Corpus 1*. Ponctuation, word stretching, repetitions, misspellings, etc. are kept as they are. For the experiments, each text is represented as a vector of words frequencies.

3. FIRST STEP: OPINION CLASSIFICATION (FROM TEXTS TO RATINGS)

The aim of this first step is to infer ratings from user reviews in order to obtain user-item-rating triples instead of user-item-review triples. This task corresponds to opinion classification. Two main approaches can be used [3]: semantinc methods which broadly consist in selecting opinion vocabulary and using it to determine texts' polarities, and statistic methods which consist in using machine learning tools. After different tries, we adopt a machine learning approach, not only for its better results, but also for the fact that it is fully automatic. We choose to use a Selective Naive Bayes (SNB) method [1]. Classification was done on two classes : positive reviews and negative reviews. F_{score} calculated on the obtained confusion matrix (table 1) is 0,71 which is not really high. This score seems to be due to the nature of texts studied : short size, misspellings, etc. These results allow to build user-item-rating matrix required for collaboratif filtering.

%	NEG	POS
$NEG	**79,08**	36,78
$POS	20,92	**63,22**

Table 1: Results of two classes classification

4. SECOND STEP: COLLABORATIVE FILTERING (FROM RATINGS TO RECOMMENDATIONS)

This second step consists in doing recommendations thanks to the results obtained with the first step. As a matter of fact, these results constitute a user-item-rating matrix which is exploitable by a recommender system based on a collaborative filtering technique [2]. Learning step of collabrative filtering consists in building similarity tables of users or items. Tool used for this experiment defines similarity between items. To do that, two measures are combined in order to obtain good results. First measure is the Pearson correlation. It corresponds to the Cosine of deviations from the mean. For this similarity measure only the set of attributes in common between two vectors is considered. Second measure is Jaccard similarity. This one measures the overlap that two vectors share with their attributes. In order to compare the different results, we compute the Root Mean Squared Error (RMSE), which is the most often used measure in recommender systems field. The RMSE measures the error rates between real and predicted ratings. It is expected to be as small as possible. Good recommender systems have a RMSE of about 0.9. The Netflix challenge was to reach a RMSE of 0.85 with their data, and this purpose required huge computations and efforts to be achieved. The recommender system described here, trained with Netflix data, obtains a RMSE of 0.91. This value will be our reference. We will compare it with RMSE obtained for predicted ratings inferred from data of other origins. The table 2 provides all results. We can notice that result obtained with infered ratings is close to Flixster's true ratings one.

	Netflix Data	Flixster Data		
	True ratings	True ratings	Infered ratings	Random ratings
RMSE	0,913	0,933	**0,936**	0,95

Table 2: Experiments' results

5. CONCLUSION

We propose a method to do recommendations from unstructured textual data in order to overcome the cold start problem. The cold start happens when the number of users registered in the service is small. Even if the idea of combining a classification step and a recommendation one is not new, this work is the first one, to our knowlege, to achieve the whole process at a large scale. It opens new perspectives for both domains. For the classification field, it validates the common intuition that opinion mining could serve as a pre-treatment for many other tasks. Recommendation is a very hot topic, and every possible way to fill a matrix of ratings without asking users to explicitely providing them is valuable. The remaining problem is that, to learn a reliable classifier, some ratings at least are necessary. But the number of examples required by a machine learning system to build a good classifier is smaller than the number of examples required by a recommender system to provide reliable recommendations. So, when ratings are harder to obtain than texts, the whole process worth being applied. For the recommendation field, our experiments first show that it is possible to feed a recommendation system with ratings coming from a completely different website. The results are not as good as with local ratings but still far better than random ones. And, if "foreign ratings" are not even available, it now remains the possibility to infer them from texts. Ratings are difficult to collect, as they are not spontaneoulsly given by users. On the contrary, blogs, forums, social networks... are quasi-infinite sources of freely available spontaneously written texts. Most of these texts very often carry opinions on subjects easy to identify. So, a large avenue of possible work seems to open.

6. REFERENCES

[1] M. Boullé. Compression-based averaging of selective naive Bayes classifiers. *Journal of Machine Learning Research*, 8:1659–1685, 2007.

[2] L. Candillier, F. Meyer, and F. Fessant. Designing specific weighted similarity measures to improve collaborative filtering systems. In *ICDM '08: Proceedings of the 8th industrial conference on Advances in Data Mining*, pages 242–255, Berlin, Heidelberg, 2008. Springer-Verlag.

[3] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2(1-2):1–135, 2008.