LSH for grouping news. Literature review

MASNA 2020-2021, Unstructured Data Analysis

Kozlova Yulia, Novoselov Mikhail

One of the fundamental problems in text mining is to examine text corpus for similar documents. This can be done in two ways - semantic similarity and content similarity. Semantic similarity is more advanced way, which assumes usage of deep-learning models such that BERT, RoBERTa, XLNet etc.(Rizvi Hasan, 19.03.2020). These models vectorize words / word sequencies according to their meaning and then compare vectors. In contrast, content similarity works as detection of word-sequence similarity of two documents and doesn't take into account the meaning of these words. Locality Sensitive Hashing (LSH) is one of content similarity methods, and further it will be discussed.

LSH is widely-spread large family of methods, which can be used in vaious text mining tasks. Different researchers and practical users suggest the following sets of task for LSH (Leskovec er al, 2020; Bawa et al., 2013; Rizvi Hasan, 19.03.2020, Aghasaryan et al., 2013):

- finding similar news articles and reprints
- mirror pages identification
- plagiarism detection
- discover similar users
- collaborative filtering - recommend to users items that were liked by other users who have exhibited similar tastes

Why LSH is so popular and so widely used? First of all, this method deals with the problem of document comparison (Bawa et al., 2013; Rizvi Hasan, 19.03.2020). In order to find similar documents, naive algorithms would compare the query document with all oher documents in the corpora. Such procedure takes a lot of time if we have, say, 10 000 news articles in the dataset. LSH algorithm would compare only documents in the same bucket, i.e. only documents which were already considered as similar. This approach greatly reduces operation time. And here, we come to another good point in favor of LSH: it resolves the curse of dimensionality. Instead of vectorising words and texts into large vectors, it creates relatively small signatures of the same length for each document, which can be compared very fastly. To sum up, LSH works very fast and that's the point.

At the same time, there are several distracting features in this method. The most important point is that it gives only approximation of nearest neigbours, i.e. most similar documents. There is always a probability that two texts, which in reality are rather similar, will be put by LSH in different buckets, and hence, the machine will not regard them as similar and will never compare them (Rizvi Hasan, 19.03.2020). Moreover, since LSH doesn't take into accoun the semantics, it ignores the fact that cognate words are quite similar. This leads to the necessity of stemming words, whic can also take quite a lot of time. In order to minimize these negative effects and achive good result, LSH can take a lot of time and memory (Ling and Wu, 2011).

Also, there is a large question about evaluating the results. In fact, LSH itself does not assume clustering the texts, it only provides nearest neighbours algorithm. So, there is no one certain way of how to evaluate these results. In the literature two approaches are suggested. Firstly, one can compare LSH similar documents with user's behavior. If a user clicked on this news or another webpage, it is regarded as similar. This was implemented in Google News personalisation algorithm (Dat et al., 2007). Secondly, one can take another operating system as ideal and compare its recommendations to LSH results (Bawa et al., 2013).

Problem of evaluating the results also leads to the question of how to use them in further analysis. In case of LSH, it usually means how to cluster documents. In its core, locality sensitive hashing does not assume clustering. However, sometimes it is suggested to take buckets as clusters (Aghasaryan et al., 2013). Moreover, different clustering algorithms can be elaborated on the basis of LSH, such K-Means, hierarchical clustering, Crest-clustering etc (Setty, 2018).

Despite the fact that LSH is a well-known and widespread algorithm, contemporary scientific research on LSH continues. Large part of the field is devoted to the usage of LSH in various recommendation systems (Qi et al., 2017), which is closely related to the modern concept of internet of things. At the same time, studying recommendation systems arises problem of user's privacy protection, which is also studied in the context of LSH-based recommendations (Chi et al., 2020; Lianyong et al., 2018). Also, there are articles devoted to technical aspects of LSH and its combinations with other methods (Shiyu et al., 2019; Youn, Shim and Lee, 2018).

References:

1. Aghasaryan, Armen & Bouzid, Makram & Kostadinov, Dimitre & Kothari, Mohit & Nandi, Animesh. (2013). On the Use of LSH for Privacy Preserving Personalization. 10.1109/TrustCom.2013.46

2. Bawa, Mayank & Condie, T. & Ganesan, Prasanna. (2005). LSH forest: Self-tuning indexes for similarity search. Proceedings of the 14th International Conference on World Wide Web. 651-660.

3. Chi, Xiaoxiao & Yan, Chao & Wang, Hao & Rafiq, Wajid & Qi, Lianyong. (2020). Amplified locality-sensitive hashing-based recommender systems with privacy protection. Concurrency and Computation: Practice and Experience. 10.1002/cpe.5681.

4. Das, Abhinandan & Datar, Mayur & Garg, Ashutosh & Rajaram, ShyamSundar. (2007). Google news personalization: Scalable online collaborative filtering. 16th International World Wide Web Conference, WWW2007. 271-280. 10.1145/1242572.1242610

5. Ji, Shiyu & Shao, Jinjin & Yang, Tao. (2019). Efficient Interaction-based Neural Ranking with Locality Sensitive Hashing. WWW '19: The World Wide Web Conference. 2858-2864. 10.1145/3308558.3313576.

6. Kang Ling and Gangshan Wu, "Frequency Based Locality Sensitive Hashing,"2011 International Conference on Multimedia Technology, Hangzhou, 2011, pp. 4929-4932, doi: 10.1109/ICMT.2011.6002015

7. Large scale document similarity search with LSH and MinHash. Rizvi Hasan. URL: https://mrhasankthse.github.io/riz/2020/03/19/Minhash-and-LSH.html

8. Leskovec, J., Rajaraman, A., & Ullman, J. (2020). Mining of Massive Datasets (3rd ed.). Cambridge: Cambridge University Press. doi:10.1017/9781108684163

9. Lianyong Qi, Xuyun Zhang, Wanchun Dou, Chunhua Hu, Chi Yang, Jinjun Chen, A two-stage locality-sensitive hashing based approach for privacy-preserving mobile service recommendation in cross-platform edge environment, Future Generation Computer Systems, Volume 88, 2018, Pages 636-643, ISSN 0167-739X, https://doi.org/10.1016/j.future.2018.02.050

10. L. Qi, X. Zhang, W. Dou and Q. Ni, "A Distributed Locality-Sensitive Hashing-Based Approach for Cloud Service Recommendation From Multi-Source Data,"in IEEE Journal on Selected Areas in Communications, vol. 35, no. 11, pp. 2616-2624, Nov. 2017, doi: 10.1109/JSAC.2017.2760458

11. Setty, Vinay. (2018). Distributed and Dynamic Clustering For News Events. 254-257. 10.1145/3210284.3219774

12. Youn, Jonghem & Shim, Junho & Lee, Sang-goo. (2018). Efficient Data Stream Clustering With Sliding Windows Based on Locality-Sensitive Hashing. IEEE Access. PP. 1-1. 10.1109/ACCESS.2018.2877138.