Content Representation and Similarity of Movies based on Topic Extraction from Subtitles

Konstantinos Bougiatiotis
Software and Knowledge Engineering Lab,
Institute of Informatics and Telecommunications,
National Center of Scientific Research
Demokritos, Greece,
bogas.ko@gmail.com

Theodoros Giannakopoulos
Computational Intelligence Lab,
Institute of Informatics and Telecommunications,
National Center of Scientific Research
Demokritos, Greece,
tyiannak@gmail.com

ABSTRACT

In this paper we examine the existence of correlation between movie content similarity and low level textual features from respective subtitles. In addition, we demonstrate the extraction of topical representation of movies based on subtitles mining. Using natural language processing and a topic modeling algorithm, namely Latent Dirichlet Allocation, applied on the movie subtitles, we extract the latent topic structure of a set of movies. In order to demonstrate the proposed content representation approach, we have built a dataset of 160 widely known movies, represented by their corresponding subtitles. After evaluating the resulting topics' quality and coherence, we move on to assert movie similarities, exploiting their distances in the topic populated space. Finally, using those topic-space projections of the movies, we aspire to create a topic model browser for movies, allowing us to explore the different aspects of similarities between movies and discover latent knowledge regarding the movies through the association of low-level topic links and high level movie similarities.

CCS Concepts

•**Information systems** → **Multimedia information systems;** *Document topic models;* •**Computing methodologies** → *Machine learning;*

Keywords

Subtitles Processing; Topic Modeling; Latent Dirichlet Allocation; Movies Similarity; Text Mining

1. INTRODUCTION

In the modern era due to the overwhelming amount of information, we need a way to filter the vast data when searching for particular information objects. This is also the case when looking at *motion pictures* in particular. With ≈340,000 feature films, not accounting for TV episodes, short films etc.[1], the different choices are, for all intents and purposes, infinite. In order to browse this huge amount of data, one needs ways of representing the movie content, as well as extracting similarities between different movies.

There are many systems providing such movie recommendation services, most of which can be classified into:

- **collaborative filtering** systems, where similarity between movies and suggestions to the user are made based on the tastes and preferences of other users who have watched the movies at hand, such as *MovieLens*[2].
- **content-based** systems, where each movie is represented by a set of features, primarily based on metadata such as, director, cast, genre etc., regarding the movie, and similarity between movies is derived by the association of these features. One of the most sophisticated such systems was *jinni*[3].
- **hybrid** systems, which are a fusion of the previous two systems such as *IMDB* and *Rotten Tomatoes*[4].

However, as we noted before, all these systems rely on human-generated information regarding the movies, in order to create a corresponding representation and assess their similarities. Even the content-based methods rely on manually produced *tags* relating to each movie (metadata), creating a database of high-level categories for classification of the movies such as *"oscar-winning", "plot twist", "master villain"* etc.. In other words, these systems do not take into account the raw content of the movie itself but solely build upon annotations made by the users.

This paper introduces the more ambitious idea of representing each movie *directly from its' content* and specifically in our approach, *from the subtitles.* We are looking for links between low-level textual similarity, extracted through the use of *Latent Dirichlet Allocation-LDA*, and high-level association of the movies, as well as new ways of representing them as mixtures of different topics, giving us the opportunity to explore the topic space of movies and discover latent knowledge about their relations.

Efforts towards the direction of using the multimedia signal of the movies, are usually limited to particular applications such as emotion extraction[13], violent content detection[10, 16], movie summarization[8] etc. An interesting

SETN '16, May 18-20, 2016, Thessaloniki, Greece

DOI: http://dx.doi.org/10.1145/2903220.2903235

[1] http://www.imdb.com/stats

[2] https://movielens.org/

[3] http://www.jinni.com/

[4] http://www.rottentomatoes.com/

application where *LDA* is used in the movie domain, aims at creating movie trailers containing movie scenes that exhibit high correlation between the subtitles present in those scenes and the movie plot as a whole[18].

The remainder of this paper is organized as follows. Firstly, the general workflow, details of the proposed method, as well as complementary techniques are explained (Section 2). We then present our data collection and ground truth generation methodology (Section 3). In the following section (Section 4), the used evaluation metrics are described and the experimental results are presented. We close by drawing conclusions and outlining topics for further research (Section 5).

2. PROPOSED METHOD

2.1 General Workflow

The overall scheme of the methodology described in the current work is presented in Figure 1. The main steps for extracting the topic representations of the movies and their similarities are the following:

- Subtitle Preprocessing: Parsing, preprocessing and vectorization of each subtitle document leading to a *bag of words* representation for each movie.
- Topic Modeling: Applying *Latent Dirichlet Allocation* on the document collection in order to train a topic model on the movies and capture their projections on the latent topic space.
- Topic Similarity: Using *cosine similarity* between each pair of movies' topic vectors, to extract similar movies as well as explore the relationship between them through the use of topics.

Let it be noted that *tf-idf* weighting and *Latent Semantic Indexing* methods have also been applied (as shown in Figure 1), for benchmarking purposes and comparison against the proposed *LDA* model.

2.2 Subtitle Preprocessing

We start by applying the preprocessing step on each subtitles document. In text mining, it is often assumed that words appear independently in a document and that their order of occurrence is immaterial for the purposes of measuring the similarities between documents or any other information retrieval task at hand. This may be a simplification, but leads to easier and faster processing of the documents. Ultimately, this assumption usually leads to the *bag of words* representation of the document, according to which each document is represented as a multi-dimensional vector. This vector is populated with the number of occurrences of the different word appearances in each document where each cell corresponds to a different word. The vector space is created by assigning a new dimension to each unique word in the document collection, that forms the *vocabulary* of the collection.

Before computing the bag of words the subtitles are filtered from irrelevant information: subtitle documents are *.srt* files with much "noise", such as timestamps and markup elements, as well as many low information words. The main steps involved in this preprocessing process are:

1. *Regular expressions removal*: filtering out timestamps etc and keeping only the subtitles
2. *Tokenization-case folding*: splitting of phrases in words and reducing all letters to lower case
3. *Lemmatization*: unifying variations of the same term due to inflectional morphology or derivationally related forms. The lemmatizer used for this purpose is based on the *WordNet* database[9]

All these text processing functionalities and many other are implemented in the Natural Language Toolkit[12] used in the present work.

As a second preprocessing stage, we apply *word filtering.* Firstly, common and movie-domain specific stopwords are removed. Then we remove words which provide low information based on how frequently they appear both in the intra-document level and the inter-document level: words occurring sparsely contain little to no information for each document and words that are frequent over the whole collection are not characteristic of a document despite their high frequency. After extensive experimentation we find the proper values of the parameters regulating the extent of filtering done to these cases.

At the end of this preprocessing set of procedures, we acquire the bag of words representation for each document, along with a corresponding unique terms vocabulary. In the sequel, we describe the three different content representation methods adopted.

2.3 Tf-idf weighting scheme

Weighting techniques are used on bag of words representations for computing the relevance of a specific word to a specific document. The most commonly used weighting technique in text mining is *term frequency-inverse document frequency (tf-idf)*[19], which accounts for the relative frequency of the term in the whole collection. It is defined as:

$$tf\text{-}idf_{i,d} = tf_{i,d} \times idf_i = tf_{i,d} \times \log_2 \frac{N}{n_i}$$

where $tf_{i,d}$ is the absolute frequency of term i in document d, N the number of documents and n_i the number of documents in our collection in which term i appears.

2.4 Latent Semantic Indexing

Although *tf-idf* is a powerful tool, more sophisticated methods to deal with synonymy, polysemy and similar situations have been proposed. These methods employ a type of principal component analysis to the document vectors, in order to reduce the dimensionality of the representation space and express the latent semantic concepts of the documents in the new space. *Latent Semantic Indexing(LSI)*[6], also called *Latent Semantic Analysis*, is such a model. It uses *singular value decomposition(SVD)* followed by *rank lowering* in order to reduce the dimensionality of the representation, thus cutting down noise in the latent space, resulting in a richer word relationship structure that reveals latent semantics present in the collection. Note that we use the LSI method after implementing the tf-idf transform on the document-word matrix of our collection. Also, the order of dimensionality reduction that LSI imposes on the word model must be decided a priori. After, extensive experimentation in our setup we selected $T = 34$ topics to be the optimal value.

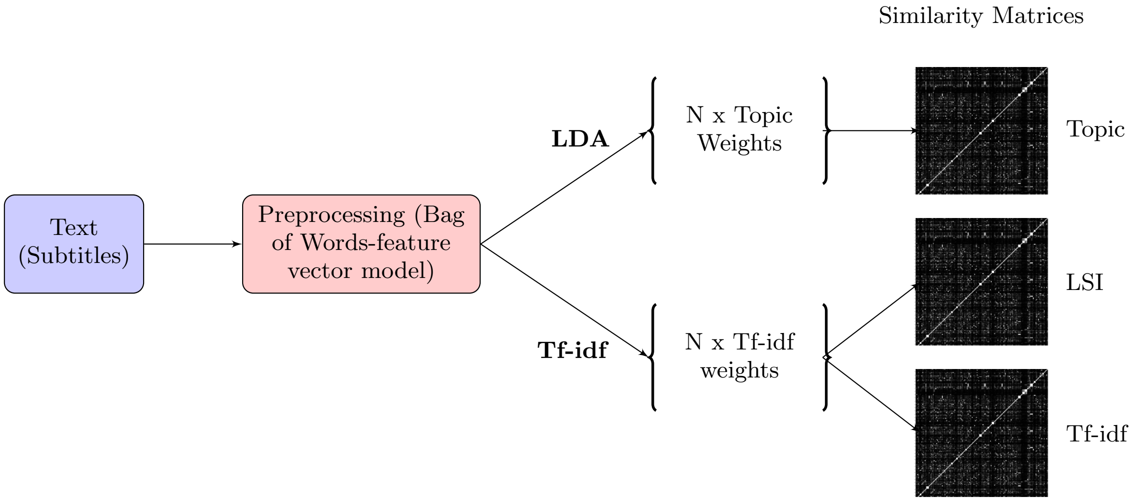

Figure 1: Workflow diagram of the proposed method

2.5 Topic Modeling

LSI provides a simple procedure for latent semantic extraction of the associations between documents. However, it faces many problems when dealing with polysemy[11], so to address this drawback we also implemented a topic model through the use of *Latent Dirichlet Allocation*(LDA)[2].

LDA is a *probabilistic topic model* where the fundamental idea is that all the documents in the collection share the same set of topics, but each document exhibits those topics in different proportions. It is a generative probabilistic process in which we assume that our samples are created through a process containing some hidden variables, the latent topic structure. Our goal when applying LDA is to infer these latent variables, the per document topic distribution and per topic word distribution, through the observed ones, namely the document-word matrix.

Maximum a posteriori estimation is intractable for this model[7], however there are many variatonal and MCMC methods for approximation of the wanted posterior. We used a *Collapsed Gibbs Sampling* version of the algorithm[14] and more specifically its implementation in the Gensim library[17](likewise for the LSI method). As is the case with LSI, we must provide LDA with the exact number of topics to be extracted. After numerous empirical evaluations on our data we settled for $T = 55$ topics to be generated.

2.6 Content Similarity

As soon as the documents are represented as feature vectors, either with tf-idf weights, LSI or topic components, we can define similarity between subtitles to correspond to the similarity of their respective representations. A popular similarity measure in text mining is *cosine similarity*, which is the angle between the vector representations of two documents:

$$CosSim(\vec{m_a}, \vec{m_b}) = \frac{\vec{m_a} \times \vec{m_b}}{\|\vec{m_a}\| \times \|\vec{m_b}\|}$$

where $\vec{m_a}, \vec{m_b}$ are the vector representations of documents a, b respectively. The range of similarity is 0-1, since all vector values are positive, with 1 denoting total similarity.

Computing the cosine similarity for each pair of movies the total *similarity matrix* is extracted. A sample similarity matrix, represented as a greyscale image, with only 6 movies (for demonstration purposes), is shown in Figure 2. This matrix was created from the topic representations of the movies, based on the LDA model. White corresponds to similarity of 1, while black to 0. In other words, the brighter a cell, the more similar the movies indexed in the corresponding row and column.

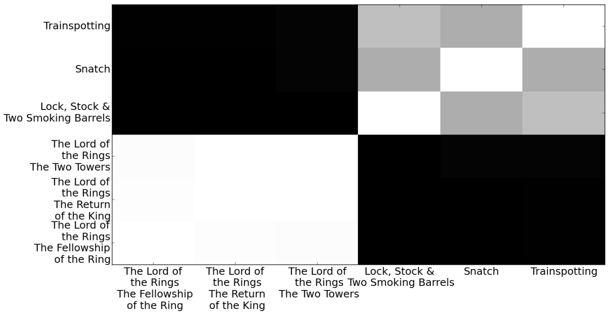

Figure 2: Sample similarity matrix with 6 movies (LDA model)

The diagonal from bottom left to top right is white, since these cells contain the self similarity of the corresponding movies. Moreover, we can see that the bottom left cells, corresponding to the movie-pairs of the *The Lord of the Rings* trilogy, are white as expected, denoting the apparent similarity between the three movies. In contrast, these three movies have no relation to the rest movies of the example (*Lock, Stock & Two Smoking Barrels, Snatch, Trainspotting*), so the respective cells are darker. Also, the movies contained in this latter set are similar to some extend, so the upper right cells are also brighter.

To further demonstrate content similarity we present Figure 3, where the topic proportions for two movies of *The Lord of the Rings* trilogy, *The Return of the King* colored in blue and *The Two Towers* colored in red, are presented. Each point in the lines denotes the *probability* (y-axis) that a word from the movies' subtitles belongs to a certain *topic* (x-axis). It is obvious that both topic distributions, though represented by 55 topics, have only "activated" very few of them. Specifically, the most active topics are #10 and #6. Examining the most probable words in topic #10 (Figure 3, upper right box) one can see that these terms (such as *ring, sam, lord, dark, precious etc*) are characteristic of the two movies and the corresponding topic is a good descriptor for both movies. The same, at a lower level, stands for topic #6. Thus, since both of them exhibit notable peaks in their topic distributions in the same topics, the cosine similarity between their topic vectors will be high.

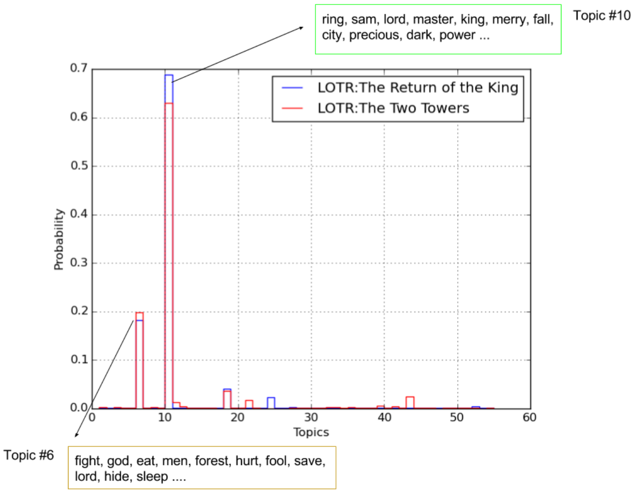

Figure 3: Topic distributions for 2 movies, inferred from a 55-topic model fitted on the dataset

3. DATASET

3.1 Data Description

In order to demonstrate the usefulness of the topical representation of the movies for similarity purposes, as well as browsing and exploration of content, we have compiled a real-world dataset of 160 movies. These movies have been selected from the *Top 250 Movies*[5] of the popular website *IMDb*. Our purpose was to use movies that are widely known and therefore we could more easily assess the quality of the results. Moreover, the dataset is populated with different types of movies to avoid metadata-specific bias, such as genre or casting. The subtitles were downloaded from an open source database[6] and were hand-checked for mistakes.

3.2 Ground-truth generation

However, to evaluate the proposed similarity representation we need a *ground-truth* similarity between the movies-documents of the dataset, against which we can pitch our results. Towards this end, we used the *Tag-genome*[21] dataset to create a ground-truth similarity matrix between the movies. According to this dataset, every movie is represented as a vector in a tag-space with ≈ 1100 unique tags and its cells are populated with a real value from 0 to 1 showing the correlation between the movie and the respective tag. The tags can be a wide variety of words-phrases such as adjectives("funny", "dark", "adopted from book"), nouns("plane", "fight") metadata("tarantino", "oscar") etc., that act as descriptors for the movies. Having this representation for each movie we calculated the cosine similarity for each possible pair of movies and obtained the ground-truth movie similarity matrix. In the sequel, this has been used as the golden standard against which our methods have been evaluated.

4. EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed topic extraction methodology, we have adopted two separate approaches. First, we carried out a simple content similarity experiment in which we evaluated the ground-truth similarity of the most similar (according to each method, tf-idf, LSI and LDA) movie. Second, we evaluated the quality of the resulting topics using intrinsic coherence measures, as well as extrinsic human evaluations.

4.1 Content Similarity

Firstly, we evaluated the ground truth similarity ranking of the most similar movie, according to each different model (tf-idf, LSI and LDA). In particular, for each different movie in the dataset we acquired the most similar movie as proposed from each different model and then we evaluated the recommendations' position with respect to the ground truth similarity matrix. Table 1 presents the true median position, calculated over all movies and ranked based on the ground truth similarity matrix, of the first most similar movie for each model. Let us note here that, we used the median of the rankings as it is more robust in skewed collections, like these of the rankings for each model.

Model	Median Ranking for 1st Recommendation
Tf-idf	18
LSI	**15.5**
LDA	**15.5**

Table 1: Median ranking of most similar movie for each model

4.2 Topic Evaluation

In this section we evaluate the quality of the topics created by the LSI and LDA Models. Traditionally in the literature, evaluation has been focused on measures based on held-out likelihood[22, 1] or an external task independent of the topic space, such as information retrieval[23]. However, it has been shown[4] that models excelling at the aforementioned measures don't necessarily generate high quality topics. For this reason, we focus on metrics that do pay attention in the resulting topic structure. The two main sub-divisions of such measures are :

- Topic diagnostic metrics: automatic diagnostic metrics based on statistics of words in topics

- Human evaluation metrics: using human judgement to examine the topic-movie relevance

Although, there is a plethora of such measures[3] we implemented a few simple ones.

[5] http://www.imdb.com/chart/top
[6] http://www.opensubtitles.org/en/search

4.2.1 Topic Diagnostic Metrics

We used two simple diagnostic metrics. The first one is an *intra-inter topic distance*. For each document in our collection, we split it in half and inferred the topic distributions in all these half parts. Then, we calculate the mean *intra-distance* and *inter-distance* as follows:

- *intra-dist* $= \sum_{i=1}^{N} CosSim(p_{i,1}, p_{i,2})$, where $p_{i,1}, p_{i,2}$ are the two halves of document i and N the total number of documents
- *inter-dist* $= \sum_{i=1}^{L} CosSim(p_{l,1}, p_{l',2})$, where l, l' are instances of L random pairs of documents and $l \neq l'$

In essence, these distances measure how semantically tight are the documents, with the higher the *intra-distance* and the lower the *inter-distance* the better because, in the first case, we want the topics on the first half of a document to be similar to topics of the second half and, in the second case, we expect different documents to exhibit different topics in general. We selected $L = 10000$ in order to have enough random pair specimens.

The second diagnostic metric is topic coherence[15], which has been shown to agree with human judgement regarding topic quality. It is defined as:

$$C(t : V^{(t)}) = \sum_{m=2}^{M} \sum_{l=1}^{m-1} log \frac{D(u_m^{(t)}, u_l^{(t)}) + 0.1}{D(u_l^{(t)})}$$

where $V^{(t)} = (u_1^{(t)}, .., u_M^{(t)})$ are the M most probable words in topic t, $D(u)$ the document frequency of term u and $D(u, v)$ the co-document frequency of terms u, v. A smoothing count of 0.1 is added, to avoid taking the logarithm of zero. Here $M = 30$.

This metric is very similar to *Pointwise Mutual Information*[5], as it is mainly motivated by word association based on the co-occurrences of the *top-M* words of each topic and the less negative it is the better the quality of the topic. It is calculated separately for each topic generated by the model and afterwards the mean topic coherence of the model is estimated.

The results of the trained topic models on these topic quality metrics are presented in Table 2. It can be clearly seen that LDA provides us with topics of better quality, as it excels over LSI in all 3 metrics presented before and specially in Topic Coherence. Hence, we move on to evaluate the content of the topics in association with the movies linked to the topic based on human judgment, for the LDA model.

Model	Intra-dist	Inter-dist	Mean Topic Coherence
LSI	0.954	0.232	-563.4
LDA	**0.959**	**0.112**	**-218.7**

Table 2: Topic quality measures for LSI and LDA models

4.2.2 Human evaluation metric

In order to evaluate the association between a movie and a topic, we devised a *topic relevance* task for the users. In this task, the subject is presented with a movie and its' most important topics, as derived from the topic weights for each movie. Each topic was represented by its' 10 most probable words. The user was asked to judge if the words presented were similar to the movie at hand, by grading them in a scale from 1 to 4, with 4 noting total relevance and 1 total irrelevance between the words and the movie. The user could also reply ”I do not know”, if for example one had not seen some of the movies. In total 10 users, both experts and non-experts on machine learning and recommendation systems, where incorporated in the task and gave input only for movies they had already seen.

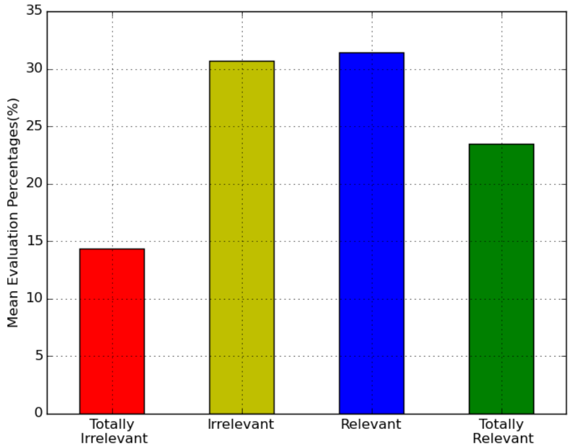

Figure 4: Human Evaluation of Movie-Topic Similarity for the LDA model

The evaluation results are shown in Figure 4. The *average standard deviation* of the evaluations was ≈ **0.81**, less than one category difference on average, meaning that the evaluations from different users are more or less correlated and they are homogenous enough for drawing conclusions. It can be seen that the distribution of votes is approximating a normal distribution, only more skewed to the right. This means that in average, most topic representations are at least relevant to the movies and the probability of a topic being totally irrelevant to a movie is half the probability of the topic being totally relevant to the movie. This makes the topic space a good representation plane for the movies.

Finally, we present some qualitative examples of our results. The following figures account for results regarding the *LDA* model. In Figure 5 we depict 4 topics generated from our movie collection, presented as word clouds were the size of each word is proportional to the importance of the word for this topic. If we observe the resulting topics, we can see that they are well formulated and coherent.

For example, the top left topic is highlighted by words such as *dad, father, mom, son, school, ...*, defining a family related topic while the bottom right exhibits mainly words *like men, colonel, war, general, ...*, defining a war related topic. Likewise, the other two word clouds define a topic about music (bottom left) and a topic about police-government (top right).

Moving on to Figure 6 we demonstrate how our topic model has clustered certain movies together based on their relevance through specific topics. In the top of the chain we

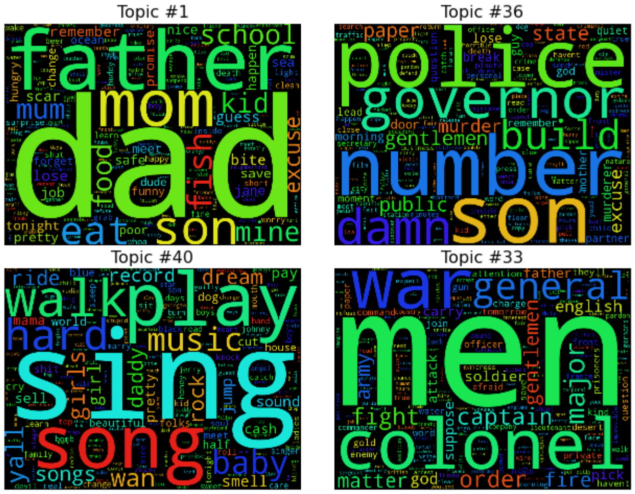

Figure 5: Word Clouds examples for 4 Topics

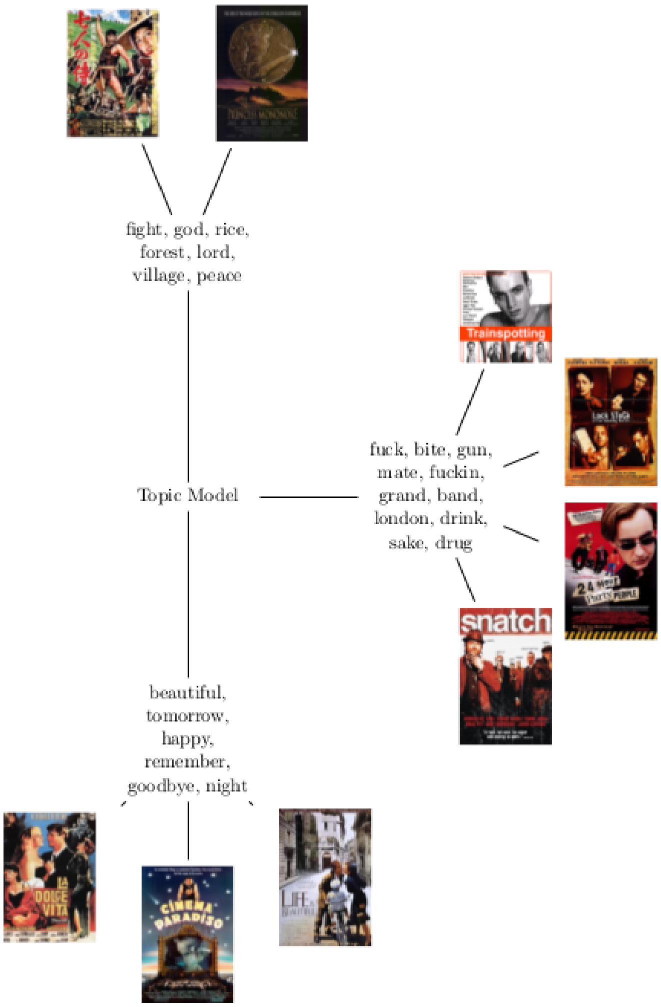

Figure 6: Topic-Movies association diagram

have the most striking example, where *Princess Mononoke* and *Seven Samurai* were grouped together. These movies are not similar according to conventional recommendation systems because one is an animation film and the other is an epic war drama. However, both are set in Japanese rural villages during feudal ages, with striking Japanese cultural elements such as strong religious beliefs, contact with nature and even consumption of rice. All these connecting details are captured in a topic whose main words are *fight*, *god*, *rice*, *forest*, *lord*, *village*, ..., as shown in the figure.

Another example is presented in the right cluster of the diagram where movies *Snatch*, *Trainspotting*, *Lock, Stock and Two Smoking Barrels* and *24 Hour Party People* are bundled. All of them are crime-comedy films from British directors that portray the British underground drug scene, containing heavily idiosyncratic British dialogues. These expressions are captured in a topic whose most probable words are *fuck*, *bite*, *mate*, *grand*, *London*, *sake*, *drug*, etc., showcasing exactly the previous information.

Likewise, in the last cluster of words Italian light comedy-drama films like *Cinema Paradiso* and *Life Is Beautiful*, with strong nostalgic expressions are put together, driven by words like *beautiful*, *happy*, *remember*, *goodbye*, *night*, etc.

5. CONCLUSIONS

In this paper, the novel idea of adopting topic representation to extract similarity of movie content has been presented. In particular, we have proposed applying topic modelling techniques on the subtitles' content. The basic outcomes of our research, as shown above, are the following:

1. A complete framework for topic extraction from movie subtitles and a method for similarity retrieval between movies based on these topics has been described in detail.

2. Experimentation has proven that *LDA* and *LSI* spaces, offer good representations for the movies, with similar results, regarding ranking recommendations. *LSI* may be preferred in terms of complexity and resources over *LDA*, however *LDA* generates topics with more human-identifiable semantic coherence, suitable for topic exploration and content similarity discovery, as shown in the qualitative examples.

3. Detailed experimental evaluation has led to the conclusion that the topic features of movies significantly correlate with the movie content and human perception of association and similarity.

4. Also, evaluation proved that through the proposed workflow, coherent and semantically compact topics are generated.

5. Showcased examples have been presented where low-level latent topic browsing can lead to knowledge discovery of high-level similarity between movies.

These results verify the core idea of this work and stimulate many future directions for our research. In particular:

- Implement scalable and efficient methods by adding more movies to our database and testing different topic models such as Hierarchical Dirichlet Processes[20] and Correlated Topic Models[1]. This will also allow us to apply proper cross-validation methods, and therefore ensure the generalization capabilities of the model.

- Experiment with audio and visual features from the movies, leading to a multimodal content based similarity system.
- Examine fusion schemes with *metadata features* (director, cast, genre, etc.) and *user preferences* (collaborative filtering) towards a hybrid system.
- Develop a web-enabled and data-driven visualization tool for topic browsing and knowledge discovery of similarities between movies.

6. ACKNOWLEDGMENTS

The authors gratefully acknowledge financial support from the Hellenic Artificial Intelligence Society (EETN) for attending this conference. Also, we are much obliged to all the annotators for donating their time and effort to provide similarity observations.

7. REFERENCES

[1] D. M. Blei and J. D. Lafferty. Correlated topic models. In *NIPS*, pages 147–154, 2005.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, Mar. 2003.

[3] J. L. Boyd-Graber, D. M. Mimno, and D. Newman. Care and feeding of topic models. In E. M. Airoldi, D. M. Blei, E. A. Erosheva, and S. E. Fienberg, editors, *Handbook of Mixed Membership Models and Their Applications*, pages 225–254. Chapman and Hall/CRC, 2014.

[4] J. Chang, S. Gerrish, C. Wang, J. L. Boyd-graber, and D. M. Blei. Reading tea leaves: How humans interpret topic models. In Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 288–296. Curran Associates, Inc., 2009.

[5] K. W. Church and P. Hanks. Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29, Mar. 1990.

[6] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, 41(6):391–407, 1990.

[7] J. M. Dickey. Multiple hypergeometric functions: Probabilistic interpretations and statistical uses. *Journal of the American Statistical Association*, 78(383):628–637, 1983.

[8] G. Evangelopoulos, A. Zlatintsi, A. Potamianos, P. Maragos, K. Rapantzikos, G. Skoumas, and Y. S. Avrithis. Multimodal saliency and fusion for movie summarization based on aural, visual, and textual attention. *IEEE Transactions on Multimedia*, 15(7):1553–1568, 2013.

[9] C. Fellbaum. *WordNet: An Electronic Lexical Database.* Bradford Books, 1998.

[10] T. Giannakopoulos, D. Kosmopoulos, A. Aristidou, and S. Theodoridis. Violence content classification using audio features. In *Proceedings of the 4th Helenic Conference on Advances in Artificial Intelligence*, SETN'06, pages 502–507, Berlin, Heidelberg, 2006. Springer-Verlag.

[11] T. L. Griffiths, J. B. Tenenbaum, and M. Steyvers. Topics in semantic representation. *Psychological Review*, 114:2007, 2007.

[12] E. Loper and S. Bird. Nltk: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*, ETMTNLP '02, pages 63–70, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[13] N. Malandrakis, A. Potamianos, G. Evangelopoulos, and A. Zlatintsi. A supervised approach to movie emotion tracking. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 2376–2379. IEEE, 2011.

[14] A. K. McCallum. Mallet: A machine learning for language toolkit. 2002. http://mallet.cs.umass.edu.

[15] D. Mimno, H. M. Wallach, E. Talley, M. Leenders, and A. McCallum. Optimizing semantic coherence in topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 262–272, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[16] J. Nam, M. Alghoniemy, and A. H. Tewfik. Audio-visual content-based violent scene characterization. In *ICIP (1)*, pages 353–357, 1998.

[17] R. Řehůřek and P. Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA. http://is.muni.cz/publication/884893/en.

[18] R. Ren, H. Misra, and J. Jose. Semantic based adaptive movie summarisation. In S. Boll, Q. Tian, L. Zhang, Z. Zhang, and Y.-P. Chen, editors, *Advances in Multimedia Modeling*, volume 5916 of *Lecture Notes in Computer Science*, pages 389–399. Springer Berlin Heidelberg, 2010.

[19] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill, 1983.

[20] Y. Teh, M. Jordan, M. Beal, and D. Blei. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006.

[21] J. Vig, S. Sen, and J. Riedl. The tag genome: Encoding community knowledge to support novel interaction. *TiiS*, 2(3):13, 2012.

[22] H. M. Wallach, I. Murray, R. Salakhutdinov, and D. Mimno. Evaluation methods for topic models. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 1105–1112, New York, NY, USA, 2009. ACM.

[23] X. Wei and W. B. Croft. Lda-based document models for ad-hoc retrieval. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, pages 178–185, New York, NY, USA, 2006. ACM.