JMLR: Workshop and Conference Proceedings vol 52, 136-146, 2016 PGM 2016

Bayesian Torrent Classification by File Name and Size Only

Eugene Dementiev E.DEMENTIEV@QMUL.AC.UK
Norman Fenton N.FENTON@QMUL.AC.UK
School of Electronic Engineering and Computer Science
Queen Mary University of London
London, UK

Abstract

Torrent traffic, much of which is assumed to be illegal downloads of copyrighted content, accounts for up to 35% of internet downloads. Yet, the process of classification and identification of these downloads is unclear, and original data for such studies is often unavailable. Many torrent items lack supporting description or meta-data, in which case only file name and size are available. We describe a novel Bayesian network based classifier system that predicts medium category, pornographic content and risk of fakes and malware based on torrent name and size, optionally supplemented with external databases of titles and actors. We show that our method outperforms a commercial benchmark system and has the potential to rival human classifiers.

Keywords: Bayesian network; causal Bayesian classifier; torrent classification.

1. Motivation and Requirements

Downloadable files called *BitTorrents* (Cohen, 2011), or just *torrents*, typically contain movies, music, games, software, books etc. and have names such as the example in Figure 1. A combined estimate provided by Cisco (2015) and Sandvine (2014) indicates that BitTorrent traffic accounts for between 5% to 35% of household and mobile traffic, depending on the region. Torrent downloads are mostly posted on the Internet by individual users, as opposed to companies, and this makes them very difficult to account for in any terms other than traffic volume. Typically, torrents have no conventional metadata or naming standard, so it is very difficult to adequately analyse and classify them reliably and autonomously.

```
{www.scenetime.com}Law.and.Order.SVU.S13E10.480p.WEB-DL.x264-mSD.exe}
```

Figure 1: Example of a Torrent Name

A report by Envisional (2011), commissioned by a large media and entertainment company NBCUniversal, claimed that two thirds of BitTorrent traffic were infringing content. A report by Tru Optik (2014) claimed that the global value of content downloaded from torrent networks amounted to more than £520 billion in 2014. This figure assumes that everything that was downloaded and presumably watched would otherwise be officially purchased. This assumption may be indicative of bias and ignores the fact that some people may have financial, geographical or language barriers to accessing material legally (Masnick, 2015). Tru Optik also provide categorised download figures, but fail to specify how the classification and identification was made, and nor does Envisional in their 2011 report.

Crucially, companies like these may have a strong vested interest in overestimating the volume and value of downloaded copyrighted content. An important drawback of such reports is the assumption that all torrent content, or even all downloaded content, is illegal. On the contrary, it is suggested that in 2014 digital music sales (i.e. legal downloads) matched revenue from physical sales (IFPI, 2015), and millions of items are available legally and for free via torrent networks (Internet Archive, 2016; Gizmo's Freeware, 2015).

Clearly, there is a motivation for being able to identify and classify such downloads automatically in a transparent and coherent way, so that unbiased statistical data can be publicly obtained.

One such classification system has been developed by a private research organisation called MovieLabs, which is funded by six major US Hollywood studios to accelerate the development of technologies essential to the distribution and use of consumer media MovieLabs (2016). MovieLabs conduct continuous in-depth research of the material posted on torrent networks. Their system (which we refer to as *MVL*) allows them to capture, organise and analyse a very large collection of torrent files posted on multiple online platforms throughout the Internet. MVL primarily aims to detect torrents that contain movies, but they also capture items of other categories and attempt to classify them by media category and as pornographic ("porn" onwards) content. The MVL system represents a viable benchmark against which a new automated approach can be tested. Given the complexity of torrent names and obfuscation involved, it is increasingly difficult for automated systems like MVL to achieve good results with a deterministic approach. Hence, a primary motivation was to develop a system that could additionally provide probabilistic classifications and be able to provide reasonable classifications even without reference to a database of known titles (this latter requirement is important because of the difficulty and cost of maintaining such databases).

Another requirement is driven by the need to identify potential fakes and malware (which includes all undesirable software such as viruses and adware). These additional requirements originate from the way the big movie studios (or companies like MediaDefender (Wikipedia, 2016) operating on behalf of the studios), attempt to combat piracy. They attempt to disrupt file sharing networks by flooding the network with multiple fakes aimed at particular high interest titles (e.g. new Hollywood blockbusters). Maintaining a high number of active fake downloads reduces the chances of a common user downloading the actual movie. This tactic minimises losses by tackling the free downloads at the time most critical for cinema ticket sales. While most fakes are either trailers or broken videos, many fakes are also posted by hackers and contain malware or employ social engineering methods to lead an unsuspecting user to web pages that may be infected.

The system described in this paper was developed, partly in collaboration with MovieLabs, to satisfy all the above requirements and is based on a Bayesian network model that integrates data and knowledge.

The rest of the paper is structured as follows: in Section 2 we describe the data used to refine our domain knowledge and compare performance of different classifying agents. Section 3 briefly introduces relevant background in Bayesian classification and covers our Bayesian model in more detail. Section 4 provides background on performance evaluation and present a comparison of results between our system, MVL and human classifiers. Section 5 summarises the paper and draws conclusions.

2. Data and Objectives

We developed a system (called *Toran* onwards) to identify and classify torrents that uses an expert-based Bayesian network that incorporates causal elements into its probabilistic model. Toran is implemented in Java and uses AgenaRisk API Agena (2016) to perform Bayesian calculations. In order to test our approach, we compare it to the benchmark commercial system MVL.

MovieLabs provided us with a large number of torrent records processed by their system, enabling us to compare our results to MVL. We studied a set of 2,500 items (called DS2500 onwards) and used it, together with expert feedback provided by MovieLabs, to build a model presented in the next section. Note that, with the exception of some prior probabilities, we did not use any machine learning approaches for two reasons. First, we could not obtain a sufficiently large sample of items where actual contents have been fully verified (downloaded and executed); and second, we wanted the model and prototype to follow human reasoning as much as possible.

Toran surpasses functionality provided by MVL by returning probabilistic predictions of not only media category and porn content, but also the risk of fakes and malware. We consider a torrent to be a fake if it was created to mislead the public rather than share its supposed original content. For example, the torrent's name from Figure 1 contains a TV series title and some information about the video quality and encoding, so it clearly attempts to communicate that the torrent contains a video file. Yet, if the torrent actually contains a broken video it is then a fake. And in case it is a malicious executable file instead, it is not only a fake, but also malware.

Data	**Interpretation**
www.scenetime.com	Tracker website
Law.and.Order.SVU	TV series “Law & Order: Special Victims Unit”
S13E10	Season 13 episode 10
480p	Video resolution
WEB-DL	Video capture source type
x264	Video encoding
-mSD	Distributor signature
exe	File type extension – software

Table 1: Data Meaning of Torrent Name Example from Figure 1

In order to be able to identify torrent items as particular movie, game or TV titles, we use the plain text data files provided by IMDb (2016). The threat of fakes and malware may be detected early, even before a file is downloaded, by analysing the file name and size, and matching it to an external database of potential threats. Table 1 provides a breakdown of the torrent name into meaningful pieces of data.

In addition to the data provided by MovieLabs and external title databases, we hired three trained human experts to manually process more than 3,000 items, including the set mentioned above. In this paper the following additional data sets are referred to:

- DS120 is a set of 121 items with fully verified contents (media categories and porn), which compares relevant predictions made by humans, MVL and Toran;
- DSFM is a set of 100 items with fully verified contents (fake, malware or none), which compares fake and malware predictions made by humans and Toran;

- DSBS is a set of 45,209 items captured from live BitSnoop tracker feed. We use this set to compare MVL and Toran media category predictions versus classifications provided by BitSnoop. It is crucial to note that DSBS has a very different category distribution to the data we originally studied when building Toran's model.

3. Bayesian Network for Torrent Classification

A Bayesian network (BN) is a probabilistic graphical model (Pearl, 1988; Fenton and Neil, 2012), consisting of variables or events, formally called nodes, connected with arrows, formally called edges and representing causal or influential relationships between the nodes. For a pair of events A and B in a relationship $A \to B$, A is called parent and B is called child. Nodes that do not have a parent are called root nodes. Associated with each child node is a conditional probability table (CPT) that captures the probabilities associated with each combination of parents' states.

The traditional method of using Bayes' theorem for classification or recommender systems is to use a very simple BN model, namely the naïve Bayes' classifier. This method assumes event independence, i.e. that attributes appear in an item completely independently from each other. While logically wrong, such classifiers were proven to be very effective (Langley et al., 1992; Sahami, 1996; Friedman, 1997), especially for text-based classification.

We extend the idea of a naïve Bayesian classifier by including some causal relationships into our model, shown in Figure 2. This BN's structure and prior probabilities combine knowledge elicited from MovieLabs experts and our own observation of data. Note that nodes with rectangular dashed border represent sub-models for diagram simplicity. The core concept is that the torrent type, also called "medium", dictates other properties of the file. However, as we may not directly observe the actual medium or other properties (green nodes), we have to predict them based on observable evidence (pink). While *Porn*, *Fake* and *Malware* nodes are Boolean, *Real Medium* has a number of labels for various file types (e.g. music, movies, books etc.).

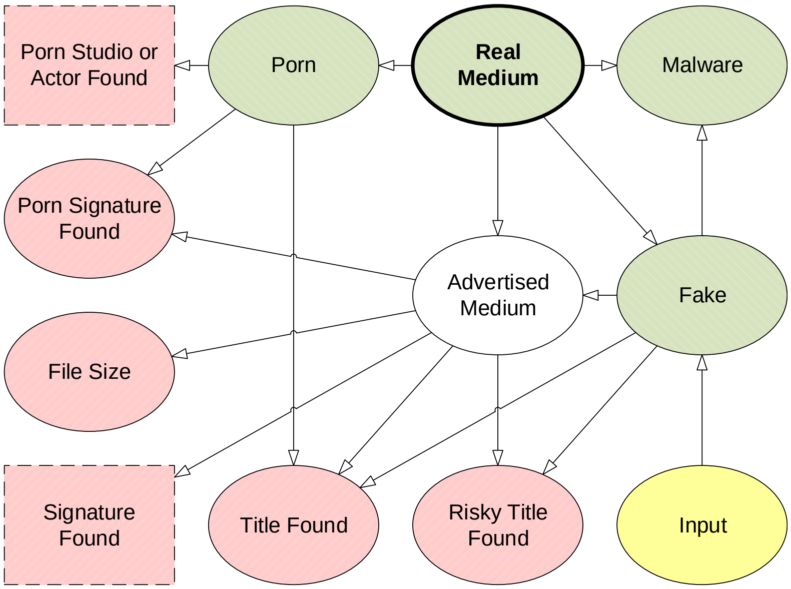

Figure 2: Basic Overview of the Current Model (dashed nodes represent sub-models)

The most important concept here is the distinction between the actual item's content and its appearance. All available evidence is based on what the item looks like, and sometimes the looks are deceiving. ***Real Medium*** is a root node and represents our prior belief about the actual category distribution of items. ***Fake*** node allows us to consider a scenario when an item is a fake, in which case its *Real Medium* is most likely to be a generic or broken video or software. This means that it may appear differently from what it really is, which is encoded in the ***Advertised Medium*** node (e.g. when *Fake* is true and item's *Real Medium* is non-movie non-TV video, it is very likely that the item is advertised as a movie; and when *Fake* is false, *Advertised Medium* simply translates *Real Medium* probabilities). ***Malware*** node specifies propensity of *Real Medium* categories to contain malicious software and defines a higher chance of malware when an item is a *Fake*. ***File Size*** is continuous and specifies most likely size distributions for each category and is a useful additional attribute.

The original expertise provided to us by MovieLabs revolved around the idea of *keywords*, which could be detected as substrings in a torrent name and associated with a medium category. We re-defined this approach in terms of *signatures*, which also consider context and whether the name string contains: specific medium category evidence, mention of a language, subtitles info, date or year. For the purpose of the BN model, a signature is evidence found in the torrent name that is associated with a particular medium category and has a strength value for this association. The ***Signature Found*** nodes are continuous and there is one for each medium category and any additional attributes such as year and date patterns, subtitles, and mention of languages. Our configuration of signatures allows many-to-many associations between categories and signatures, which is very useful when a signature is shared between multiple categories. For example, `HDTVRIP` is most relevant for "Video: TV" category, as well as, to a lesser extent, for "Video: Movie", but not other categories. In addition, the ***Porn Signature Found*** node is defined such that some supposedly porn keywords can be found for non-porn items. Most prominent examples are music file names that often contain substrings such as `Sexy Trance`, `VA Sex4yourears` and others. ***Porn Studio or Actor Found*** are two Boolean nodes and simply encode an observation that a known studio or actor was detected in the name string.

Categories can also be associated with each other, which gives us the ability to capture complex relationships in the CPTs for evidence-signature-detected nodes. First, an item that belongs to a category A is expected to have strong evidence associated with A in the name string. Second, presence of weak evidence associated with category A may contribute to our belief that the item actually belongs to a *different* category B. For example, weak music and movie evidence may increase the probability of the item actually being a TV series or episode.

While parsing a torrent name for signatures, we attempt to filter out as many characters as safely as possible so that, ideally, only the item's title remains. We then match it to a list of game, movie and TV series titles using a sequence alignment algorithm (Smith and Waterman, 1981). Though this is an optional piece of evidence, a high quality and well maintained database can improve results considerably. *Title Found* node captures the observation that the name string matched positively to a title of a particular category. As indicated earlier, upcoming, recently released or otherwise relevant movie titles are often aggressively protected by their owners and are targeted by hackers. Before a movie is leaked or digitally released to the public in any other way (e.g. on DVD, Blu-ray, official web download etc.), its title should be listed as "risky". We notice that after a movie is digitally released, it is always guaranteed to be available on the Internet, and may soon be moved from the "risk group" to the general title database. *Risky Title Found* captures whether the item was positively matched to such a "risky" title.

Prior probabilities and CPTs for different nodes are based on:

- *Real Medium* is based on original estimates from MovieLabs in combination with independent assessment of DS2500 by the three experts;
- *Advertised Medium* is driven by expert feedback and aims to capture the intent behind sharing a fake item;
- *Fake*, *Malware* and *Porn* are based on our study of verified files in separate small datasets;
- *(Risky) Title Found* and *Porn Actor or Studio Found* are driven by performance of the Toran's text matching component;
- *(Porn) Signature Found* and *File Size* are based on data from DS2500 and corrected by us to ensure they were not overfitted.

The full model, as well as all relevant datasets, are available online (Dementiev, 2016). The BN can be opened with AgenaRisk Free (Agena, 2016). Note that for practical reasons some continuous nodes were discretised manually, although AgenaRisk supports dynamic discretisation.

4. Empirical Evaluation

This section covers relevant evaluation background, classifying agents' output compatibility and provides the actual results comparison between the human panel *HP*, *MVL*, *Toran* and *ToranT* (with title matching turned on and off respectively). Note that each item was classified by instantiating values of the pink nodes from Figure 2 and performing propagation with AgenaRisk's implementation of sum-product algorithm (Pearl, 1982) in conjunction with dynamic discretisation (Neil et al., 2007).

4.1 Error Metrics

Probabilistic predictions are made in the form of a prediction vector, which is composed of probabilities assigned to each of the possible classes, such that all of them sum to 1. For example, if there are only 3 possible classes *A*, *B* and *C*, the prediction vector S^P may be {0.7, 0.2, 0.1}, which may be interpreted as a high probability that the item belongs to class *A*. For compatibility, the actual state of an item S^R is a similar vector where a single class has a weight of 1.

We use a combination of *Brier score* (BS) and *absolute error* (AE) to evaluate the quality of probabilistic predictions. These metrics were selected because they address different aspects of classification analysis.

Brier score (Brier, 1950) puts a higher emphasis on the spread of the wrong portion of a prediction, while absolute error is more concerned with the confidence of correct portion of the prediction. BS is equal to the sum of squared differences between predicted and actual probability values for each class or state and is defined as:

$$BS(S^R, S^P) = \sum_{c \in C} (S^P_c - S^R_c)^2, 0 \leq BS \leq 2$$

where $BS(S^R, S^P)$ is the Brier score, calculated for a pair of probability vectors, C is the set of possible classes, c is a particular class, S^R_c is the true state proportion of the item within category c and S^P_c is the predicted proportion of the item within c. Higher values of BS correspond to worse predictions, because it is an error metric, while 0 is indicative of a perfect prediction.

Absolute error (Armstrong, 2001; Hyndman and Koehler, 2006) considers only the absolute difference between the prediction and the actual state. When the true state is a single category, AE is equal to the difference between 1 and the predicted probability of the item to belong to the correct category, hence 0 denotes a perfect match while 1 means a complete mismatch. AE is defined as:

$$AE(S^R, S^P) = 1 - P(S^P_{c_A} = S^R_{c_A}), 0 \leq AE \leq 1$$

where c_A denotes the actual category of the item. Average AE over a set of multiple items is called *mean average (percentage) error* (MAE).

4.2 Agent Output Compatibility

Output formats provided by the classifying agents vary greatly. While humans and MVL select a single sub-category or super-category and may return an “Unknown” result, Toran always returns a probabilistic distribution across all sub-categories. In one type of validation we formatted hard human panel votes and MovieLabs decisions as probability distributions, which would be compatible with results returned by Toran or any other probabilistic system. In our taxonomy there are 6 super-categories and 13 total sub-categories.

Category	Vote Counts		Derived Prior Distributions			
	Humans	MVL	“Unknown”		Partial	
			Human	MVL	Human	MVL
Audio: Music	971	35	0.15	0.03	0.94	1.00
Audio: OST	29	0	0.00	0.00	0.00	0.00
Audio: Other	59	0	0.01	0.00	0.06	0.00
Image	187	55	0.03	0.04	1.00	1.00
Mixed	0	0	0.00	0.00	1.00	1.00
Software: Game	271	51	0.04	0.04	0.31	0.57
Software: Other	604	42	0.09	0.03	0.69	0.43
Text: Book	279	17	0.04	0.01	0.80	1.00
Text: Magazine	69	0	0.01	0.00	0.20	0.00
Text: Other	24	0	0.00	0.00	0.00	0.00
Video: Movie	1,732	564	0.27	0.46	0.45	0.55
Video: Other	966	129	0.15	0.11	0.25	0.13
Video: TV	1,174	330	0.18	0.27	0.30	0.32
Partial	*563*	*1,033*				
Unknown	*572*	*244*				

Table 2: Medium Prior Distributions for Humans and MVL Based on DS2500

Every hard sub-category verdict given by a human expert or MVL can be represented as a vector of probabilities S^P with the single selected label having a probability of 1 and all other labels having a probability of 0. However, when a super-category or "Unknown" was chosen, we map it to the sub-category vector according to the agent's prior. In order to define these priors, we calculated classification decisions of humans and MVL on a separate data set. Table 2 shows category vote counts and derived prior distributions in general and within super-categories. For example, if MVL classifies an item as "Video", the relevant prediction probability vector has all sub-categories set to 0, except for video sub-categories, which are set according to how MVL generally selects video sub-categories.

In DS2500 MVL classified 1033 items with only a super-category and 244 as "Unknown". Between the three human classifiers these figures were 563 and 572 respectively.

4.3 Accuracy Metrics

In addition to evaluating probabilistic prediction error, we calculate accuracy of hard choices made, based on probabilistic predictions. For example, the top predicted category may be selected as the hard choice and compared to the actual category. We calculate values of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN) and derive from them the associated values of:

Recall (ρ), also called *sensitivity*, which is the number of correct positive classifications out of the number of actual positives, and is defined as:

$$\rho = \frac{TP}{TP + FN}, 0 \leq \rho \leq 1$$

Precision (π), also called *positive predictive value*, which is the proportion of correct instances of classification out of the whole set of positive decisions performed, and is defined as:

$$\pi = \frac{TP}{TP + FP}, 0 \leq \pi \leq 1$$

Matthews correlation coefficient (MCC) (Matthews, 1975), which uses all classification results, unlike ρ and π that were criticised by Powers (2011) for not using false negatives. MCC is considered to be more balanced and is defined as:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}, -1 \leq MCC \leq 1$$

4.4 Classification Results Comparison

We employ 95% confidence intervals to calculate margins of error based on standard error of the mean.

		Error		Accuracy		
Set	**Agent**	**BS**	**MAE**	ρ	π	**MCC**
DS120	HP	0.30 ±0.08	0.38 ±0.05	0.80	0.80	0.79
	MVL	0.64 ±0.13	0.43 ±0.08	0.56	0.56	0.53
	Toran T	0.46 ±0.10	0.40 ±0.06	0.66	0.66	0.63
DSBS	MVL	0.384 ±0.006	0.298 ±0.004	0.72	0.72	0.696
	Toran	0.269 ±0.005	0.231 ±0.003	0.811	0.811	0.796
	Toran T	0.268 ±0.005	0.217 ±0.003	0.816	0.816	0.800

Table 3: Average Error and Accuracy per Agent for Medium (DS120 and DSBS)

Table 3 demonstrates the improved accuracy of Toran compared to MVL irrespective of the metrics used. A bigger difference in BS compared to MAE between MVL and Toran is due to AE accumulated by Toran even for correct predictions (e.g. a movie classified correctly with 0.87 probability still gets 0.13 error score). Although we may not be able to draw strong conclusions based on DS120 due to its size, DSBS clearly demonstrates that Toran outperforms MVL even on data that Toran is not specifically attuned to (very different category priors and DSBS has fewer categories that are present at all). Interestingly, humans are not as accurate as we expected them to be and demonstrate at least 30% error. This means that algorithms trained on purely human-processed data will inevitably inherit such a handicap.

		Error	Accuracy		
	Agent	**MAE**	ρ	π	**MCC**
Porn	HP	0.13 ±0.03	0.94	0.97	0.94
	MVL	0.21 ±0.07	0.31	1.00	0.49
	Toran T	0.19 ±0.04	0.61	0.92	0.67
Fakes	HP	0.38 ±0.08	0.30	0.64	0.19
	Toran T	0.29 ±0.08	0.57	0.84	0.51
Malware	HP	0.17 ±0.07	0.26	1.00	0.46
	Toran T	0.18 ±0.07	0.30	0.88	0.45

Table 4: Average Error and Accuracy per Agent for Porn (DS120), Fakes and Malware (DSFM)

Table 4 shows that MVL and Toran are similarly accurate at predicting porn content with Toran being slightly better, and humans performing much better than automatic systems. Most importantly, however, we see that Toran is a viable system for pre-emptive fakes and malware detection, as it rivals humans in malware prediction and outperforms in fakes detection (MVL has no capability to determine fakes and malware). While both DS120 and DSFM are relatively small samples, they still establish viability of the system. All data sets and results are available in full online (Dementiev, 2016).

5. Summary and Conclusions

We have explained the need for improved methods of torrent identification and classification, and the limitations of current deterministic systems that attempt to do so. The Toran system we described is based on a causal Bayesian network model that extends traditional Bayesian classifier methods. Toran provides improved accuracy over the commercial benchmark system (MVL) and also provides analyses that are beyond the scope of deterministic systems. Moreover, Toran is capable of operating successfully even in full autonomous mode without using an external title database and selecting the state with highest probability as a hard classification. Medium and porn classification results show clearly that a maintained title database is not required to achieve a considerable improvement over the benchmark system. The prototype system still relies on non-Bayesian components (such as the string analysis and title matching) and we believe that it could achieve even better results with improvements made to these components. The core of the BN model is based on the idea of advertised content, which matches the actual content when the original poster intends to simply share the item. However, if the original intent is to deceive, the actual unwanted or malicious content may masquerade as an item that is appealing to a regular user. This core BN concept can be reused in other domains, such as online retail product classification, and search query parsing and is already being used in attribution of artwork.

Acknowledgments

This work was supported by the EPSRC (PhD research) and in part by ERC Advanced Grant ERC-2013-AdG339182-BAYES_KNOWLEDGE. We also thank Agena Ltd for access to its API software and the anonymous referees for their comments.

References

Agena. Download AgenaRisk, 2016. URL `http://www.agenarisk.com/products/`.

S. Armstrong. Evaluating Forecasting Methods. In *Principles of Forecasting: a Handbook for Researchers and Practitioners*, chapter 14, pages 443–472. 2001. ISBN 978-0-306-47630-3. doi: 10.1007/978-0-306-47630-3_20.

G. W. Brier. Verification of Forecasts Expressed in Terms of Probability. *Monthly Weather Review*, 78:1–3, 1950. doi: 10.1175/1520-0493(1950)078⟨0001:vofeit⟩2.0.co;2.

Cisco. Forecast and Methodology, 2014-2019, 2015.

B. Cohen. The BitTorrent Protocol Specification, 2011. URL `http://www.bittorrent.org/beps/bep_0003.html`.

E. Dementiev. Toran Online, 2016. URL `http://toran.som-service.com/`.

Envisional. An Estimate of Infringing Use of the Internet. Technical report, Envisional, 2011.

N. Fenton and M. Neil. From Bayes' Theorem to Bayesian Networks. In *Risk Assessment and Decision Analysis with Bayesian Networks*, chapter 6.2, pages 132–134. CRC Press, 2012. ISBN 978-1439809105.

J. H. Friedman. On Bias , Variance , 0 / 1 — Loss , and the Curse-of-Dimensionality. *Data Mining and Knowledge Discovery*, 1(1):55–77, 1997. ISSN 13845810. doi: 10.1023/A:1009778005914.

Gizmo's Freeware. 30 Sites For Legal (and Free) Torrents, 2015. URL `http://www.techsupportalert.com/content/finding-legal-and-free-torrents.htm`.

R. Hyndman and A. Koehler. Another Look at Measures of Forecast Accuracy. *International Journal of Forecasting*, 22(4):679–688, 2006. doi: 10.1016/j.ijforecast.2006.03.001.

IFPI. IFPI Digital Music Report 2015. Technical report, IFPI, 2015.

IMDb. Alternative Interfaces, 2016. URL `http://www.imdb.com/interfaces/`.

Internet Archive. Archive Torrents, 2016. URL `https://archive.org/details/bittorrent`.

P. Langley, W. Iba, and K. Thompson. An Analysis of Bayesian Classifiers. In W. Swartout, editor, *Proceedings of the Tenth National Conference on Artificial Intelligence*, number 415 in AAAI'92, pages 223–228. AAAI Press, 1992. ISBN 0262510634.

M. Masnick. MPAA's Lies About Films Being Available Online Easily Debunked In Seconds, 2015.

B. Matthews. Comparison of the Predicted and Observed Secondary Structure of T4 Phage Lysozyme. *Biochimica et Biophysica Acta (BBA) - Protein Structure*, 405(2):442–451, 1975. doi: 10.1016/0005-2795(75)90109-9.

MovieLabs. Motion Pictures Laboratories, Inc., 2016. URL `http://www.movielabs.com/`.

M. Neil, M. Tailor, and D. Marquez. Inference in Hybrid Bayesian Networks using Dynamic Discretisation. *Statistics and Computing*, 17(3):219–233, 2007.

J. Pearl. Reverend Bayes on Inference Engines: A Distributed Hierarchical Approach. In *Proceedings of the Second National Conference on Artificial Intelligence (AAAI-82)*, pages 133–136, Pittsburgh, PA, 1982. AAAI Press.

J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers Inc., 1988. ISBN 978-1558604797.

D. Powers. Evaluation: From Precision, Recall and F-Measure To ROC, Informedness, Markedness & Correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011. doi: 10.1.1.214.9232.

M. Sahami. Learning Limited Dependence Bayesian Classifiers. In *KDD96 Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pages 335–338. AAAI Press, 1996.

Sandvine. Global Internet Phenomena Report, 2014.

T. F. Smith and M. S. Waterman. Identification of Common Molecular Subsequences. *Journal of Molecular Biology*, 147:195–197, 1981. doi: 10.1016/0022-2836(81)90087-5.

Tru Optik. Digital Media Unmonetized Demand and Peer-to-Peer File Sharing Report. Technical report, Stamford, CT, 2014.

Wikipedia. MediaDefender, 2016. URL `https://en.wikipedia.org/wiki/MediaDefender`.