O

Opinion Stream Mining

Myra Spiliopoulou[1], Eirini Ntoutsi[2,3], and Max Zimmermann[4]
[1]Otto-von-Guericke University-Magdeburg, Magdeburg, Germany
[2]Leibniz Universität Hannover, Hanover, Germany
[3]Ludwig Maximilians Universität München, Munich, Germany
[4]Swedish Institute of Computer Science (SICS Swedish ICT), Kista, Sweden

Abstract

Opinion stream mining aims at learning and adaptation of a polarity model over a stream of opinionated documents, i.e., documents associated with a polarity. They comprise a valuable tool to analyze the huge amounts of opinions generated nowadays through the social media and the Web. In this chapter, we overview methods for polarity learning in a stream environment focusing especially on how these methods deal with the challenges imposed by the stream nature of the data, namely the nonstationary data distribution and the single pass constraint.

Work partially done while with the Ludwig-Maximilians University, Munich.

Synonyms

Mining a Stream of Opinionated Documents; Polarity Learning on a Stream

Definition

Opinion stream mining is a variant of stream mining, of text mining and of opinion mining. Its goal is learning and adaptation of a polarity model over a stream of opinionated documents. An "opinionated document" is a text associated with a "polarity." Polarity is a value that represents the "strength" and the "direction" of an opinion. The strength can be a categorical value (e.g., $+$, $-$) or a ranking value (e.g., zero to five stars) or a continuous value (e.g., in the interval $[0, 1]$). The direction refers to whether the opinion is positive, negative, or neutral. Strength and direction are often mixed. For example, in a ranking using stars, five stars may stand for a very positive opinion, zero stars for a very negative one, and three stars for a neutral one.

As a variant of stream mining, opinion stream mining is subject to challenges of learning on a stream: adapting to changes in the data generating distribution – a phenomenon often called *concept drift* and processing the data as they arrive (in a single pass), since they cannot be retained permanently.

As a variant of text mining, opinion stream mining is subject to challenges of learning from texts: identifying the parts of speech that

© Springer Science+Business Media New York 2016
C. Sammut, G.I. Webb (eds.), *Encyclopedia of Machine Learning and Data Mining*,
DOI 10.1007/978-1-4899-7502-7_905-1

are in the text (e.g., verbs, adjectives, etc.); bringing the individual words into stem form (e.g., "opinions"→"opinion"); deciding which words will constitute the feature space and which are not informative and should be ignored; modeling the similarity between texts, taking (among other issues) differences in the length of texts into account; extracting the "entities" from the text (e.g., persons, products); and detecting the "topics" of discourse in the texts.

As a variant of opinion mining, opinion stream mining faces further challenges: distinguishing between words that bear sentiment (e.g., "nice," "ugly") and those referring to facts (e.g., "sauna," "phone") and discerning different forms of sentiment (e.g., anger, joy). For static data, these challenges are addressed with techniques of natural language processing (NLP), text mining, and **Sentiment Analysis and Opinion Mining** (cf. lemma).

The aforementioned challenges are exacerbated in the stream context. Opinion stream mining provides solutions for learning and adapting a polarity model in a volatile setting: the topics in the opinionated documents may change; the attitude of people toward an entity (e.g., person, product, event) may change; the words used by people to express polarity may change; and even the words used by people, i.e., the vocabulary, may also evolve over time.

Motivation, Main Tasks, and Challenges

With the rise of WEB 2.0, more and more people use social media to upload opinions on essentially every subject – on products, persons, institutions, events, and topics of discourse. These accumulating opinionated data are valuable sources of information that can deliver valuable insights on the popularity of events; on the properties of products that are deemed important; on the positive or negative perception people have toward a product, person, or institution; on their attitude toward a specific subject of discourse; etc.

Background: The analysis of opinionated data is investigated in the research areas of *sentiment analysis* and *opinion mining*. These two areas overlap, whereby research on sentiment analysis puts more emphasis in understanding different types of "sentiment" (e.g., irony, anger, sadness, etc.), while opinion mining focuses more on learning models and discerning trends from data that simply have positive or negative "polarity" (or are neutral). For an extensive discussion of the subject, the reader is referred to the lemma **Sentiment Analysis and Opinion Mining**.

In Liu (2012), Bing Liu defines four opinion mining tasks as follows:

1. *Entity extraction:* "Extract all entity expressions in a document, and categorize or group synonymous entity expressions into entity clusters. Each entity expression cluster indicates a unique entity e_i."
2. *Property extraction:* "Extract all property expressions of the entities, and categorize these property expressions into clusters. Each property expression cluster of entity e_i represents a unique property a_{ij}."
3. *Opinion holder extraction:* "Extract opinion holders for opinions from text or structured data and categorize them. The task is analogous to the above two tasks."
4. *Sentiment classification:* "Determine whether an opinion on a property a_{ij} is positive, negative, or neutral, or assign a numeric sentiment rating to the property."

Among these tasks, the first one is not peculiar to opinion mining: *entity extraction* (EEX) is a subtask of document analysis. A widespread special case of EEX is named-entity recognition (NER); a minister is an entity, and a specific minister is a named entity. The goal of EEX and NER is to identify and annotate all entities in a document. To this purpose, NLP techniques are used, as well as collections of "named entities"; a list of the towns in a country is an example of such a collection.

The second task can be generalized in two ways. First, the properties need not be associated to an explicitly defined entity (e.g., a person or city); they may also be topics or subtopics under a subject of discourse (e.g., air pollution as a subtopic of environment pollution). Further, clustering is not the only way of identifying properties/topics: aspect-based opinion mining is a subdomain of topic modeling (cf. lemma **Topic Models for NLP Applications** for the general domain). In this subdomain, a document is perceived as a mixture of topics and sentiments.

In opinion *stream* mining, the collection of opinionated documents is not perceived as a static set but as an ongoing stream. While the first and third of the aforementioned tasks remain largely unchanged, the second and forth task must be redefined in the stream context. The task of property extraction on the stream is addressed with methods of *dynamic topic modeling* (see Blei and Lafferty (2006) for the core concepts) and with methods of text stream clustering (Aggarwal and Yu 2006).

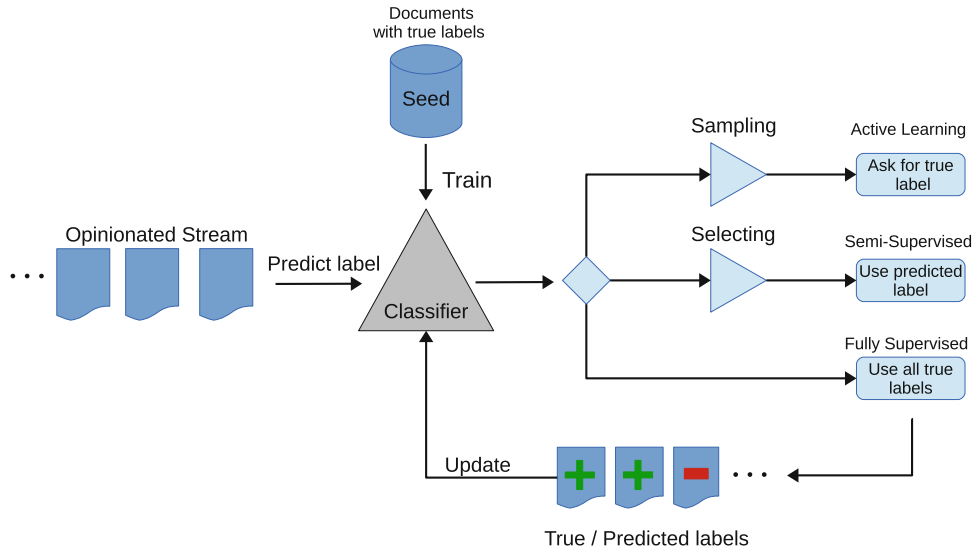
The task of sentiment classification becomes a stream classification problem for an evolving text stream. Hereafter, we denote this task as "learning a polarity model" or simpler "polarity model learning," without referring explicitly to the fact that the model is learned on a stream.

Challenges of opinion stream mining: The challenges faced in opinion stream mining for property extraction and polarity learning emanate from the different aspects of volatility in the opinionated stream:

(a) *The data evolve with respect to the target variable:* The attitude of people toward a subject of discourse, a person, a product, or some property of this product may change over time. This corresponds to a change in the priors of the polarity class.
(b) *The topics evolve:* New subjects of discourse emerge, some product properties become uninteresting while others gain momentum. The learning algorithm must recognize that people discuss different topics.
(c) *The vocabulary evolves:* New words show up, some words fall out of use, and the polarity of some words may change. This means that the high-dimensional feature space used by the learning algorithm changes during the process of learning and adaption.
(d) *Labels are scarce:* In conventional stream classification, it is assumed that fresh labels are timely available for classifier adaption. Opinionated streams are fast and the inspection of opinions is a tedious task. So, the demand for human intervention/supervision for document labeling must be limited.

Main tasks of opinion stream mining: In response to challenges (a) and (c), opinion stream mining encompasses solutions for polarity model learning and adaption and also when the class priors change and when the vocabulary evolves. Next to fully supervised solutions, there are also semi-supervised learning methods and active learning methods, in response to challenge (d). In the following, we elaborate on supervised, semi-supervised, and active stream mining approaches for the classification of opinionated streams.

For challenge (b), we refer the reader to literature on text stream clustering, starting, e.g., with Aggarwal and Yu (2006), and to literature on dynamic topic modeling, starting with Blei and Lafferty (2006) and Wang and McCallum (2006). Dynamic topic modeling for opinionated document streams gained momentum in the last years, resulting in several works on dynamic topic mixture models that capture both aspects (properties) and sentiment. An example is Fu et al. (2015) on dynamic nonparametric hierarchical Dirichlet process topic modeling. An important characteristic of this work is that the number of topics can be determined automatically and adjusted over time. Further, an aging (time-decay) component is incorporated into the learning process; this allows for forgetting old topics (Fu et al. 2015). As we discuss in the next section, the issue of forgetting is also essential in supervised learning over the stream, as means of adaptation to concept drift.



Opinion Stream Mining, Fig. 1 Polarity learning on a stream of opinionated documents – fully supervised, semi-supervised, and active learning options

Polarity Learning in an Opinionated Stream

Polarity learning is a supervised task that involves model learning and model adaption over an opinionated stream, i.e., an infinite sequence D of arriving *instances* $d_1, \cdots, d_i, \cdots$. An instance/opinionated document is a vector over a *word vocabulary* V, which is built up and changes over time.

An instance has a polarity label c. We denote the class attribute by C. Much of the research on opinion stream mining considers streams where documents have positive or negative polarity and are mixed with neutral documents. We use this convention in the following, i.e., we assume that the polarity label is one of positive (+), negative (−), or neutral (Ø).

Workflow

The fully supervised stream learning scenario implies that the model is continuously learned on arriving, labeled instances. To deal with the label scarcity challenge, opinion stream mining research also contributes semi-supervised methods that learn with only an initial seed of labeled instances and active learning methods that request a label for only some of the arriving instances. An abstract workflow of the learning tasks is depicted in Fig. 1, distinguishing among supervised, semi-supervised, and active learning.

As can be seen in the figure, an initial classifier is trained on a starting set of manually labeled instances *Seed*. This set can be a small corpus of carefully selected opinionated documents that are representative of the stream, at least at the beginning, or the *Seed* can consist solely of the first arriving documents in the stream. Labels delivered by a human expert are denoted in the figure as "true labels," as opposed to the "predicted labels" that are assessed by the classifier.

In each subsequent step, the classifier predicts the labels of the arriving documents. For supervised learning, a human expert immediately delivers the true labels, which are then used for model adaption. In semi-supervised learning, the classifier adapts by using (a selection of) instances with predicted labels. In active learning, the expert is asked to deliver true labels only for some of the arriving documents which are then used for model adaption. These three ways of polarity learning are discussed hereafter.

The instances of the stream may be processed one by one as they arrive, or they may be

stored into "chunks" (also called "blocks" or "batches"). In the first case, i.e., in "instance-based" processing, the classifier is adapted after seeing each new instance. In "chunk-based" processing, the classifier adapts after each chunk. A chunk may be a fixed-sized block of documents or it may be defined at different levels of temporal granularity, e.g., hourly, daily, or weekly. Instance-based processing allows for fast adaption; however, the processing cost is higher as the model is updated after each instance. Chunk-based processing is more appropriate for streams where changes in the topics and/or vocabulary are manifested gradually. A detailed discussion of instance- vs chunk-based methods can be found in the lemma **Stream Classification**.

Fully Supervised Opinion Stream Classification

Fully supervised polarity learning on an opinionated stream is performed in the same way as stream classification in a conventional stream. The reader is referred to the lemma **Stream Classification** for a detailed elaboration on the interaction between the classifier and the stream, the detection of drift, and the adaption of the model. For opinionated streams, two aspects are of particular interest: how to choose a classification algorithm for polarity learning and how to deal with changes in the vocabulary.

Stream classification algorithms for polarity learning. Since there are many stream classification algorithms, it is reasonable to investigate how appropriate they are for learning on an opinionated stream. Several comparative studies have emerged at the beginning of the decade, including Bifet and Frank (2010) and Gokulakrishnan et al. (2012). In Gokulakrishnan et al. (2012), Gokulakrishnan et al. study a Twitter stream (i.e., a stream of short texts) and evaluate multinomial Naive Bayes (MNB), support vector machines (SVM), Bayesian logistic regression, sequential minimal optimization (SMO), and random forests (RF); they show that Bayesian classifiers, RF, and SMO outperform the other methods. In Bifet and Frank (2010), Bifet et al. compare MNB, stochastic gradient descend (SGD), and a Hoeffding tree (HT) algorithm; they report that MNB and SGD perform comparably when the stream is stable, but MNB has difficulties in adapting to drifts. In terms of efficiency, MNB is the fastest and HT is the slowest.

In their survey on concept drift adaption (Gama et al. 2014), Gama et al. elaborate on how *forgetting* of old data can be used to adjust a model to drift, and they discuss different forgetting strategies. The Hoeffding tree variant AdaHT (Bifet and Gavaldà 2009) forgets subtrees if performance degrades. In an opinionated stream, it is reasonable to also forget *words*, i.e., parts of the feature space, since the choice of words used in the data (here: documents!) may also change. The MNB variant proposed in Wagner et al. (2015) quantifies the contribution of a word to the polarity model by considering the number of documents containing this word and the recency of these documents; this variant is shown to adapt well to changes in the stream.

Stream classification algorithms for an evolving vocabulary. The problem of vocabulary evolution is rarely investigated in the context of stream mining. There are studies on online topic modeling and clustering on text streams, in which the model is adapted when the vocabulary – the feature space – changes (AlSumait et al. 2008; Gohr et al. 2009; Zimmermann et al. 2016), but most studies assume that all words are known in advance, and only their contribution to the model may change over time.

Among the stream classification algorithms, adaption to an evolving vocabulary is possible for some algorithms. The Hoeffding tree variant AdaHT (Bifet and Gavaldà 2009) can forget deprecated words when it forgets parts of the model (subtrees) and may be able to include new words when it builds new subtrees. The multinomial Naive Bayes variant proposed in Wagner et al. (2015) does modify the vocabulary, by considering at each timepoint only words that appear often in recent documents.

Adaption to an evolving vocabulary is an open problem. Currently, only few stream classification algorithms can deal with changes in the

feature space. How to employ other classification algorithms over the opinionated stream? The fallback solution is to extend the workflow by a task that regularly recomputes the vocabulary/feature space from the most recent documents and then re-initializes the polarity model. This solution has the disadvantage that the old model is completely forgotten, but the advantage that any stream classification algorithm can be used for learning.

Semi-supervised Opinion Stream Classification

Goal of semi-supervised stream learning is to learn a model on an initial set of manually labeled documents, sometimes called the "seed set" or "initial seed," and then adapt the model by using the arriving unlabeled instances. Semi-supervised methods have the inherent advantage of not demanding human intervention after the initialization of the model.

For this family of methods, the initial seed is the only available ground truth. Hence, it is essential that the instances comprising the seed set are a representative sample. Evidently, this sample ceases being representative, as soon as concept drift occurs. Semi-supervised learning algorithms adapt to drift by building a training set that consists of the initial seed and arriving unlabeled instances, to which they themselves assign the labels. There are two strategies for the selection of unlabeled instances to be labeled by the classifier and added to the training set. The first strategy chooses instances on the grounds of the classifier's confidence to the predicted labels. The second strategy chooses instances by considering their similarity to previously labeled instances.

First strategy. Chapelle et al. point out that "Probably the earliest idea about using unlabeled data in classification is self-learning, which is also known as self-training, self-labeling, or decision-directed learning. This is a wrapper-algorithm that repeatedly uses a supervised learning method. It starts by training on the labeled data only. In each step a part of the unlabeled points is labeled according to the current decision function; then the supervised method is retrained using its own predictions as additional labeled points . . . " (Chapelle et al. 2006). However, self-training may lead to performance deterioration, because erroneous predictions of the classifier lead to erroneous labels in the training set.

Another approach is the "co-training" of several independent classifiers (Blum and Mitchell 1998). In the context of text classification, Aggarwal and Zhai propose to split the feature space into subsets and train an independent classifier on each subset (Aggarwal and Zhai 2014); then, high-confidence predictions of each single classifier are used to feed the other classifiers with new labels, so that no classifier is trained on its own predictions.

An example of co-training on a stream of tweets is in Liu et al. (2013): the complete feature space encompasses both text features (such as adjectives) and non-text features (e.g., emoticons). Views are built over this feature space, and a classifier (multiclass SVM) is trained on each view, using a small set of labeled instances only.

Second strategy. As an alternative to self-training and co-training, the second semi-supervised strategy adds to the training set those instances that are most similar to already labeled instances. One way to capitalize on labeled instances under this strategy is to cluster labeled and unlabeled instances together, then determine the label of each cluster from the labeled instances in it, and finally select for training some unlabeled instances per cluster (e.g., those closest to the cluster center).

In the context of opinionated semi-supervised stream learning, a clustering-based strategy brings two advantages. First, text stream clustering algorithms can be used, whereupon the clusters are updated gradually, as new unlabeled instances arrive. Further, these clusters reflect the properties/topics in the opinionated stream, thus addressing challenge (b) of task 2 on opinionated streams (cf. section on "Motivation, Main Tasks, and Challenges"). Example methods have been proposed by Gan et al. (2013) and by Zimmermann et al. (2015a).

In the previous section on fully supervised learning, we point out that forgetting (old data,

part of the model, part of the feature space) may be beneficial for model adaption (cf. Gama et al. 2014). When learning in a semi-supervised way, though, forgetting may have negative side effects: since the seed set is the only ground truth provided by the human expert, forgetting those "precious" data labels is likely to lead to performance deterioration.

Active Learning for Opinion Stream Classification

Similarly to semi-supervised approaches, active learning methods attempt to learn and adapt to the ongoing stream without demanding a label for each arriving instance. Instead of re-acting to the labels that become available, active methods *proactively* (thereof the name "active") request labels for the instances expected to be most informative for learning.

In active stream learning, there are two ways of requesting labels for some of the arriving instances. In the pool-based scenario, unlabeled instances are collected into a pool; the active learning algorithm chooses a subset of them and asks for their labels. In the sequential scenario, the algorithm decides for each arriving instance whether it will request a label for it. An overview of active learning methods for conventional streams is in Zliobaite et al. (2011).

Active learning is often used for various text mining tasks, including sentiment classification (Zhou et al. 2013). Active algorithms for opinionated streams also gain momentum. CloudFlows is a cloud-based platform for opinion stream mining that adheres to the pool-based scenario (Saveski and Grcar 2011; Kranjc et al. 2015): a first model of the stream is learned from a large corpus of tweets that contain emoticons; after initialization, the stream is partitioned into chunks, and an active learning algorithm is used to select instances and store them in a pool. The instances in the pool are ranked, and the top-ranked positions are shown to human experts. This approach has the advantage that human experts (e.g., in crowdsourcing) label the opinionated documents shown to them offline, whereupon these newly labeled instances are used for classifier adaption.

The algorithm ACOSTREAM (Zimmermann et al. 2015b) adheres to the sequential scenario, in the sense that sampling is done for each instance individually at its arriving time. This algorithm uses a variant of multinomial naive Bayes for classification, which (as in Wagner et al. 2015) deals with changes in the vocabulary of the arriving documents.

The multiclass active learning algorithm of Cheng et al. (2013) combines uncertainty and likelihood sampling to choose instances that are close to the current decision boundary, as well as instances from a yet unseen part of the data space. This algorithm (which adheres to the sequential scenario) is particularly interesting for learning on text streams, where some of the most recent instances may belong to an area of the data space that did not contain any instances in the past.

Recent Developments

Opinion stream mining builds upon advances in opinion mining, stream classification, active stream learning, and semi-supervised stream learning. Traditional methods in this domain have not been designed with big data in mind. However, opinionated streams have big data characteristics: volume, variability, variety, and veracity.

Volume refers to the huge number of opinions uploaded daily in social media and to the high dimensionality of the opinionated documents.

Variability refers to changes in the data flow rate and to changes in the data distribution, i.e., to concept drift.

Variety refers to the heterogeneous data types, including plain texts, images, and videos. The graph structure of the social networks, in which opinion holders are linked to each other, also adds to the variety of the data relevant to opinion mining.

Veracity refers to the uncertainty of the polarity labels provided by the human experts: labeling an opinionated stream is an inherently difficult task, since some opinionated documents

(e.g., documents containing subtle irony) may be perceived differently by different people.

Challenges associated to these four Vs are not always peculiar to opinion stream mining: while challenges associated to variability are exacerbated in the opinion stream mining context, challenges associated to, e.g., volume can benefit from general-purpose big data solutions. These include, among others, scalable machine learning and online NLP algorithms, crowdsourcing approaches for data labeling, visualization advances, and visual analytics for the monitoring and interpretation of activities on social platforms.

Open Problems

Opinion stream mining is a rather young area. Open problems include:

- How to extend the traditional notion of "concept drift" so that it also cover changes in the feature space? How to design algorithms that detect such changes and adapt to them in an efficient way?
- How to distinguish between concept drift and "virtual drift" (Gama et al. 2014), i.e., between changes that do affect the decision boundary and changes that do not?
  Especially in an opinionated stream, many changes occur at each moment, e.g., new words appear, and the number of postings changes with the hour of the day, but not all of them require model adaption. How to design algorithms that recognize virtual drift and only adapt the model when true concept drift occurs?
- How to capture changes in the semantics and polarity of words?
  If a word's semantics or polarity change, how to inform existing resources (e.g., lexica like SentiWordNet) that a word's meaning and polarity are different for old documents than for recent ones?
- How to deal with label veracity in the stream?
  A promising approach is crowdsourcing, s is done, e.g., in CloudFlows (Kranjc et al. 2015). *Amazon Mechanical Turk* is a popular platform, where one can upload tasks for crowdsourcing. However, crowdsourcing has not been designed for learning and adaption on a fast stream, so solutions that also deal with stream velocity are necessary.

An associated open issue that can also be found in text stream mining, e.g., in the analysis of news streams, concerns the description of *bursts*. A burst is a rapid increase in social activity and may also be associated with a rapid change in the class priors and in the words being used to express polarity and to express facts. Do these changes disappear after the burst fades out, or do people take up the new words/expressions and use them also when they express opinions on other subjects? Does a burst lead to (more) permanent changes in the way people express opinions, on their perception toward a given entity, or on the topics they discuss?

Impact

Opinions have been always important for decision making. The opinion deluge we encounter nowadays mainly due to the WWW and the widespread usage of social networks is transforming business, society, and our own decisions on, e.g., what product to buy, which movie to watch, etc. Opinion (stream) mining offers solutions for automatically exploiting such sort of data for decision making, through, e.g., prediction models. Beyond its usage as a "standalone tool" for, e.g., polarity prediction, opinion (stream) mining has an impact on other areas of research, an example of which is the area of *recommenders:* next to the ratings typically used by recommenders, it is possible to also capitalize on the user reviews as more and more users also provide reviews on the rated items. These reviews are rich in information: they typically describe the aspects of the items that the users like/dislike. Further, if there are no ratings, they

may be inferred from the reviews. A recent work in this area is McAuley and Leskovec (2013).

Cross-References

- Active Learning
- Concept Drift
- Co-training
- Incremental Learning
- Online Learning
- Semi-supervised Learning

Recommended Reading

Some of the publications cited thus far elaborate on issues that were only briefly touched in this lemma. In Liu (2012), Bing Liu gives a thorough overview of sentiment analysis and opinion mining. For text classification methods, readers are referred to the recent book chapter of Aggarwal and Zhai (2014).

References

Aggarwal CC, Yu PS (2006) A framework for clustering massive text and categorical data. In: Proceedings of 6th SIAM international conference on data mining (SDM'06), Bethesda. SIAM, pp 479–483

Aggarwal C, Zhai C (2014) Text classification. In: Aggarwal C (ed) Data classification: algorithms and applications, chapter 11. Chapman & Hall/CRC, Boca Raton, pp 287–336

AlSumait L, Barbara D, Domeniconi C (2008) On-line LDA: adaptive topic models for mining text streams with applications to topic detection and tracking. In: Proceedings of 2008 IEEE conference on data mining (ICDM'08), Pisa. IEEE, pp 373–382

Bifet A, Frank E (2010) Sentiment knowledge discovery in Twitter streaming data. In: Proceedings of the 13th international conference on discovery science (DS'10), Canberra. Springer, pp 1–15

Bifet A, Gavaldà R (2009) Adaptive learning from evolving data streams. In: Proceedings of the 8th international symposium on intelligent data analysis: advances in intelligent data analysis VIII (IDA), Lyon. Springer, pp 249–260

Blei DM, Lafferty JD (2006) Dynamic topic models. In: Proceedings of 23rd international conference on machine learning (ICML'06), Pittsburgh, pp 113–120

Blum A, Mitchell T (1998) Combining labeled and unlabeled data with co-training. In: Proceedings of 11th conference on computational learning theory, Madison. ACM, pp 92–100

Chapelle O, Schölkopf B, Zien A (2006) Semi-supervised learning. MIT, Cambridge

Cheng Y, Chen Z, Liu L, Wang J, Agrawal A, Choudhary A (2013) Feedback-driven multiclass active learning for data streams. In: Proceedings of 22nd international conference on information and knowledge management (CIKM'13), San Fransisco, pp 1311–1320

Fu X, Yang K, Huang JZ, Cui L (2015) Dynamic non-parametric joint sentiment topic mixture model. Know-Based Syst 82(C):102–114

Gama J, Žliobaitė I, Bifet A, Pechenizkiy M, Bouchachia A (2014) A survey on concept drift adaptation. ACM Comput Surv 46(4):44:1–44:37

Gan H, Sang N, Huang R, Tong X, Dan Z (2013) Using clustering analysis to improve semi-supervised classification. Neurocomputing 101:290–298

Gohr A, Hinneburg A, Schult R, Spiliopoulou M (2009) Topic evolution in a stream of documents. In: SIAM data mining conference (SDM'09), Reno, pp 378–385

Gokulakrishnan B, Priyanthan P, Ragavan T, Prasath N, Perera A (2012) Opinion mining and sentiment analysis on a Twitter data stream. In: Proceedings of the 2012 international conference on advances in ICT for emerging regions (ICTer), Colombo, pp 182–188

Kranjc J, Smailovic J, Podpecan V, Grcar M, Znidarsic M, Lavrac N (2015) Active learning for sentiment analysis on data streams: methodology and workflow implementation in the ClowdFlows platform. Inf Process Manag 51(2):187–203

Liu B (2012) Sentiment analysis and opinion mining. Synth Lect Hum Lang Technol 5(1):1–167

Liu S, Li F, Li F, Cheng X, Shen H (2013) Adaptive co-training SVM for sentiment classification on tweets. In: Proceedings of 22nd international conference on information and knowledge management (CIKM'13), San Fransisco, pp 2079–2088

McAuley J, Leskovec J (2013) Hidden factors and hidden topics: understanding rating dimensions with review text. In: Proceedings of 7th ACM conference on recommender systems (RecSys'13), Hong Kong. ACM, pp 165–172

Saveski M, Grcar M (2011) Web services for stream mining: a stream-based active learning use case. In: Proceedings of the workshop "Planning to Learn and Service-Oriented Knowledge Discovery" at ECML PKDD 2011, Athens

Wagner S, Zimmermann M, Ntoutsi E, Spiliopoulou M (2015) Ageing-based multinomial naive bayes classifiers over opinionated data streams. In: European conference on machine learning and principles and practice of knowledge discovery in databases (ECMLPKDD'15), Porto, 07–11 Sept 2015. Volume 9284 of lecture notes in computer science. Springer International Publishing

Wang X, McCallum A (2006) Topics over time: a non-Markov continuous-time model of topical trends. In: Proceedings of 12th ACM SIGKDD international conference on knowledge discovery and data mining (KDD'06), Philadelphia, pp 424–433

Zhou S, Chen Q, Wang X (2013) Active deep learning method for semi-supervised sentiment classification. Neurocomputing 120:536–546

Zimmermann M, Ntoutsi E, Spiliopoulou M (2015a) Discovering and monitoring product features and the opinions on them with OPINSTREAM. Neurocomputing 150:318–330

Zimmermann M, Ntoutsi E, Spiliopoulou M (2015b) Incremental active opinion learning over a stream of opinionated documents. In: WISDOM'15 (workshop on issues of sentiment discovery and opinion mining) at KDD'15, Sydney

Zimmermann M, Ntoutsi E, Spiliopoulou M (2016) Extracting opinionated (sub)features from a stream of product reviews using accumulated novelty and internal re-organization. Inf Sci 329:876–899

Zliobaite I, Bifet A, Pfahringer B, Holmes G (2011) Active learning with evolving streaming data. In: Proceedings of ECML PKDD 2011, Athens. Volume 6913 of LNCS. Springer