BaggTaming
Learning from Wild and Tame Data

Wikis, Blogs, Bookmarking Tools - Mining the Web 2.0 Workshop

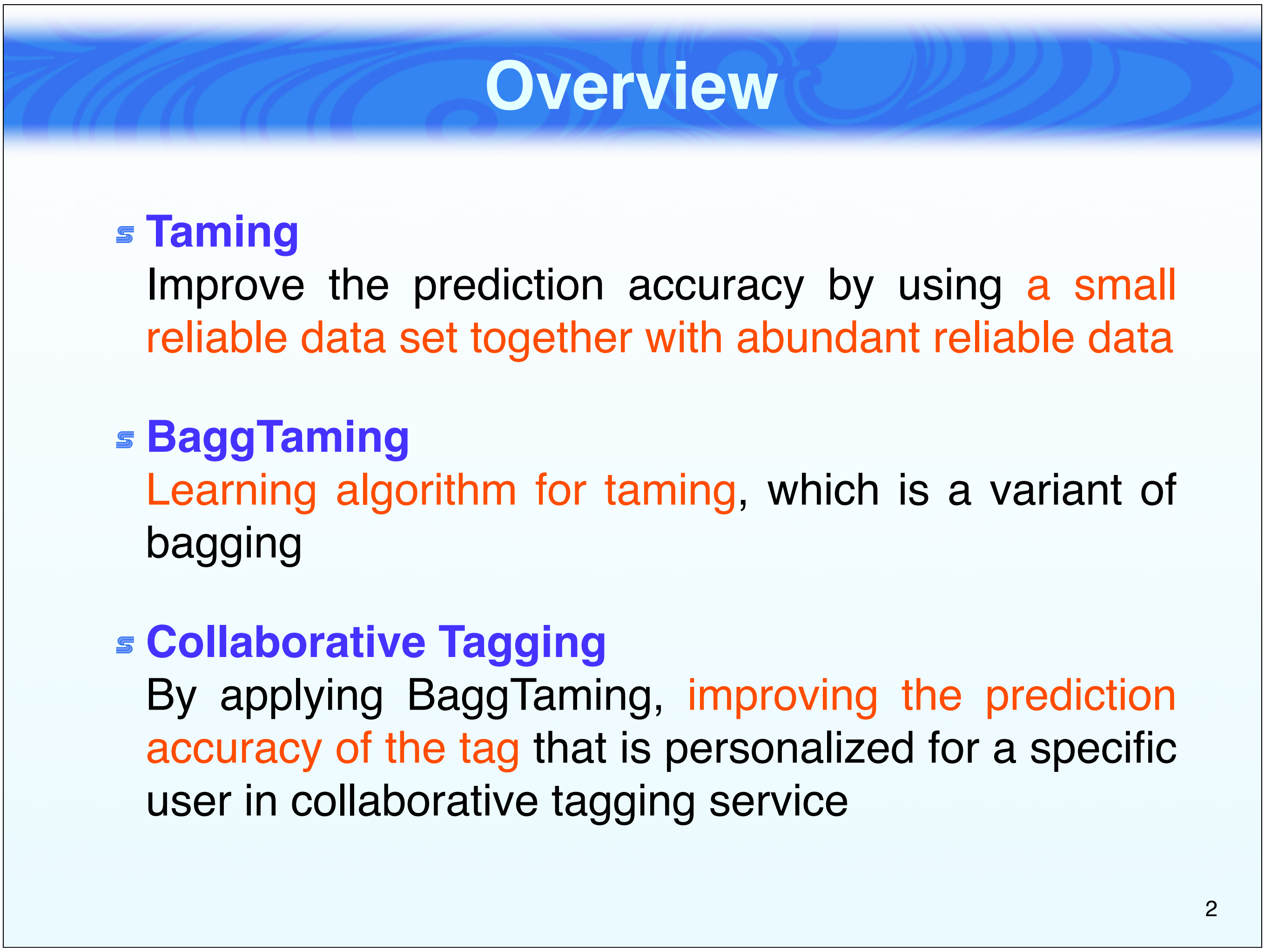
@ECML/PKDD2008 Workshop, 15/9/2008

Toshihiro Kamishima, Masahiro Hamasaki, and Shotaro Akaho

National Institute of Advanced Industrial Science and Technology (AIST)

http://www.kamishima.net/

Today, I'd like to talk about a new learning framework and its application to the prediction tag personalization.



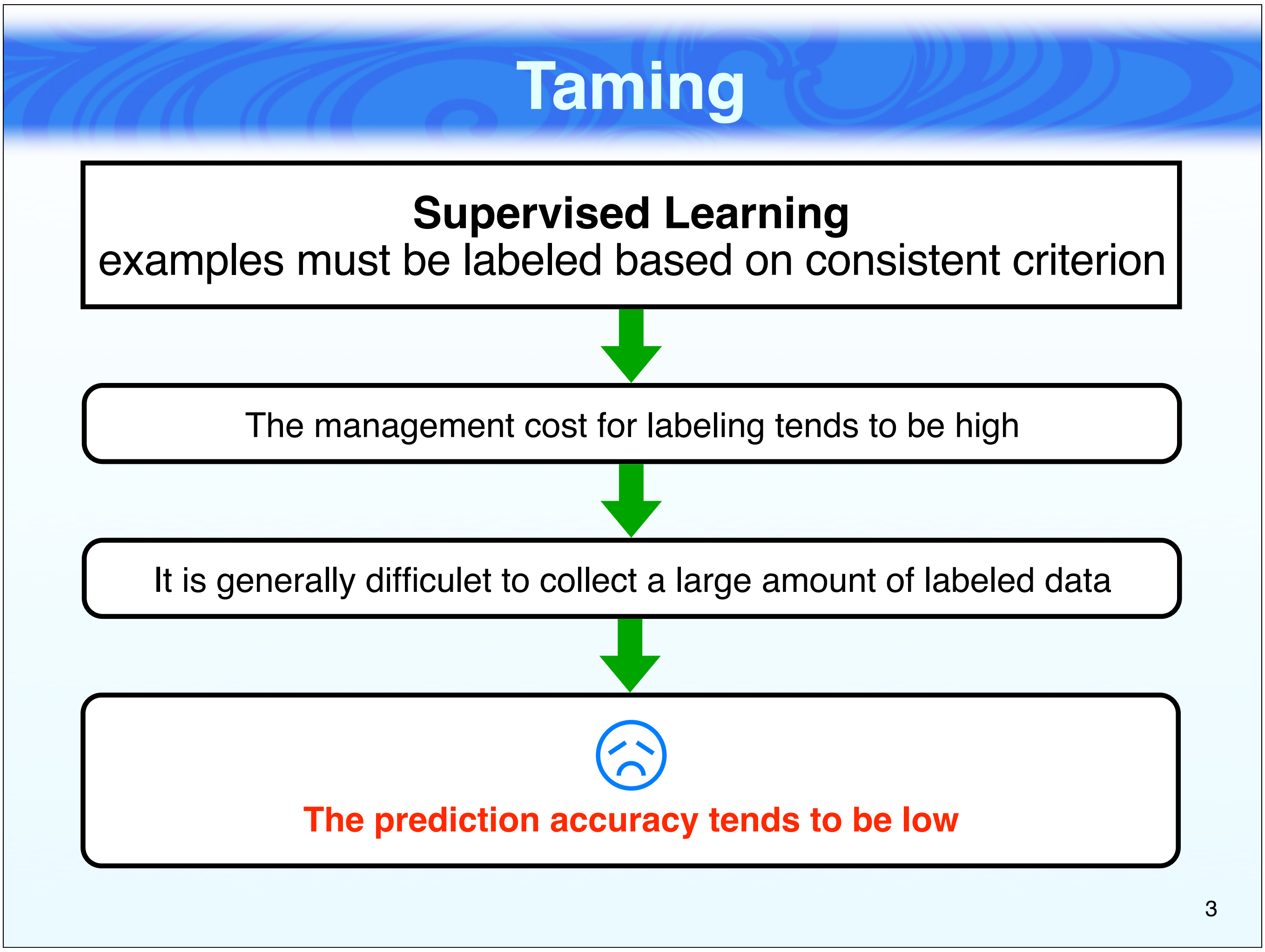
Overview

- **Taming**
  Improve the prediction accuracy by using a small reliable data set together with abundant reliable data

- **BaggTaming**
  Learning algorithm for taming, which is a variant of bagging

- **Collaborative Tagging**
  By applying BaggTaming, improving the prediction accuracy of the tag that is personalized for a specific user in collaborative tagging service

We first talk about a learning framework, Taming. This framework uses two types of data sets, tame and wild.
Next, we developed a BaggTaming algorithm for this framework, that is a variant of a bagging.
Finally, we apply this BaggTaming to the tag personalization task for the social bookmarking service.



Taming

Supervised Learning
examples must be labeled based on consistent criterion

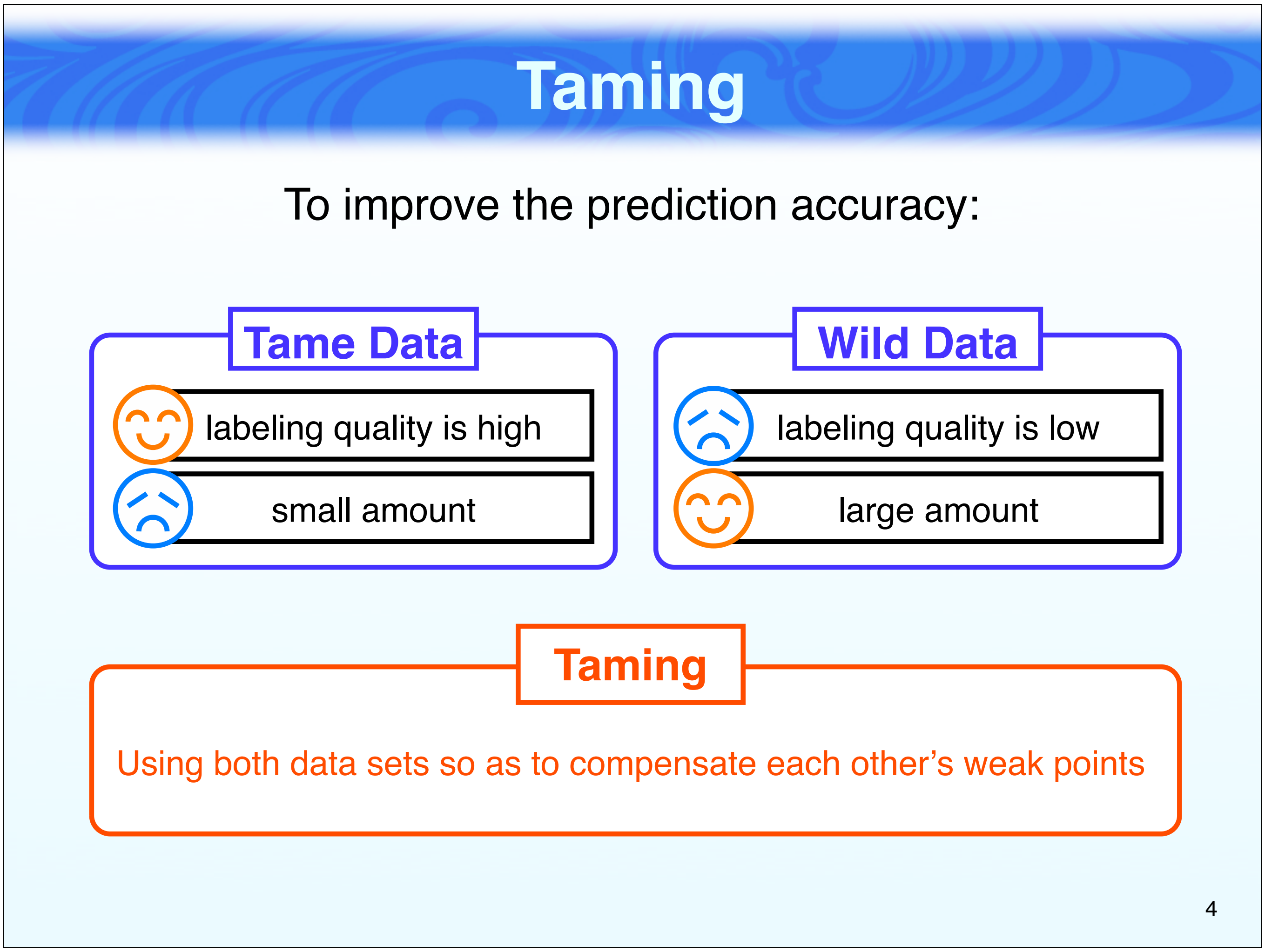
The management cost for labeling tends to be high

It is generally difficulet to collect a large amount of labeled data

☹

The prediction accuracy tends to be low

We begin with a machine learning framework, taming.
Supervised learning requires training examples that are labeled as consistent as possible with the target concept.
Keeping such consistency is laborious, so the management cost for labeling tends to be high.
Therefore, it is generally difficult to collecting a large amount of labeled data.
Consequently, the prediction accuracy tends to be low.



Taming

To improve the prediction accuracy:

Tame Data

☺ labeling quality is high

☹ small amount

Wild Data

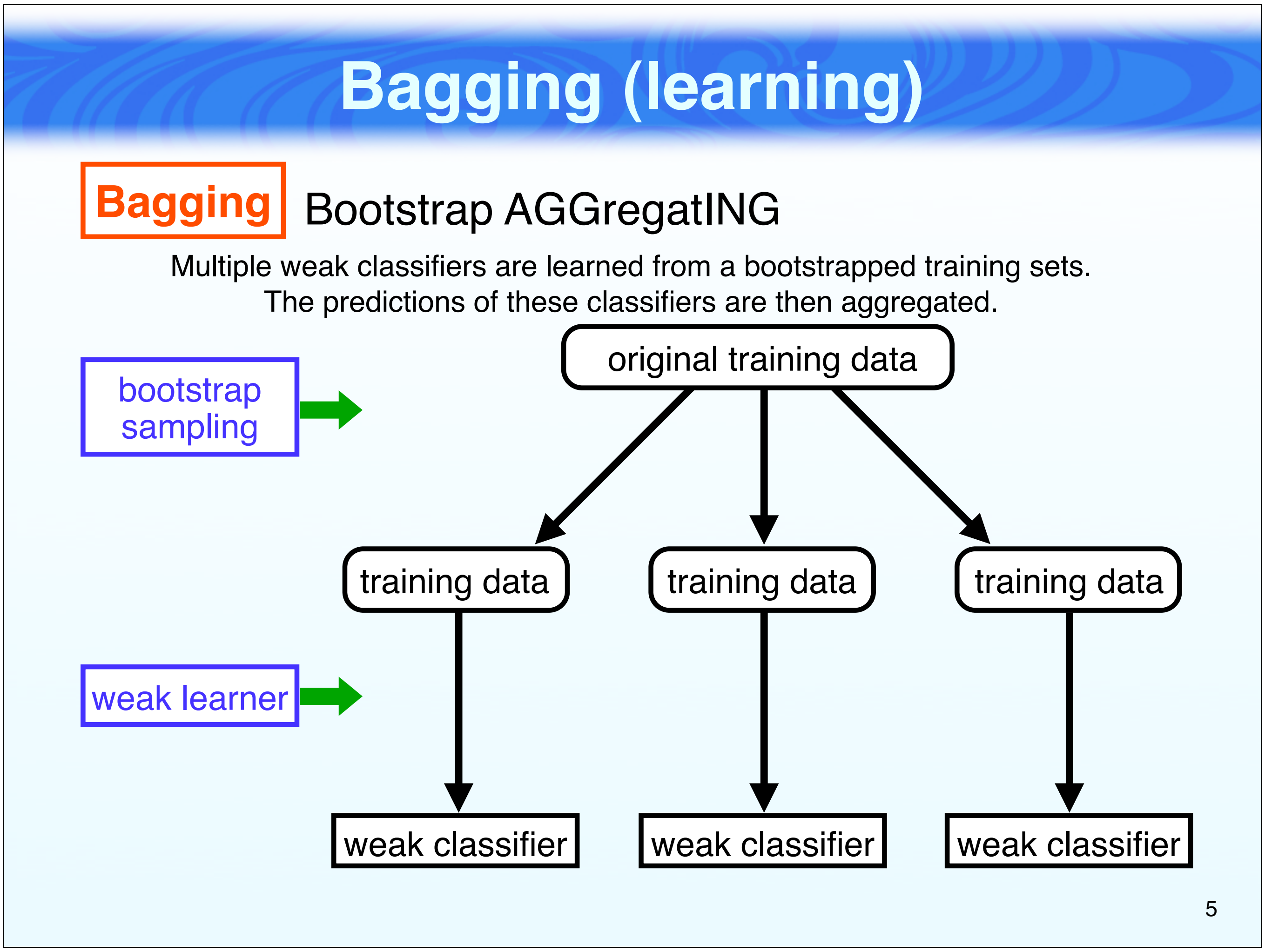
☹ labeling quality is low

☺ large amount

Taming

Using both data sets so as to compensate each other’s weak points

To relieve this difficulty, we propose a learning framework, taming.
We employed two types of training data sets, tame and wild.
The tame data are carefully labeled and so the labeling is highly consistent with the target concept. However, due to its labeling cost, a relatively small amount of data are available.
On the other hand, the labeling quality of wild data is low, but these data are much more abundant.
By using both data sets so as to compensate each other’s weak points, we try to improve the prediction accuracy.
Next, we will talk about a BaggTaming algorithm that is designed for taming.



Bagging (learning)

Bagging Bootstrap AGGregatING

Multiple weak classifiers are learned from a bootstrapped training sets.
The predictions of these classifiers are then aggregated.

bootstrap sampling

weak learner

original training data

training data

training data

training data

weak classifier

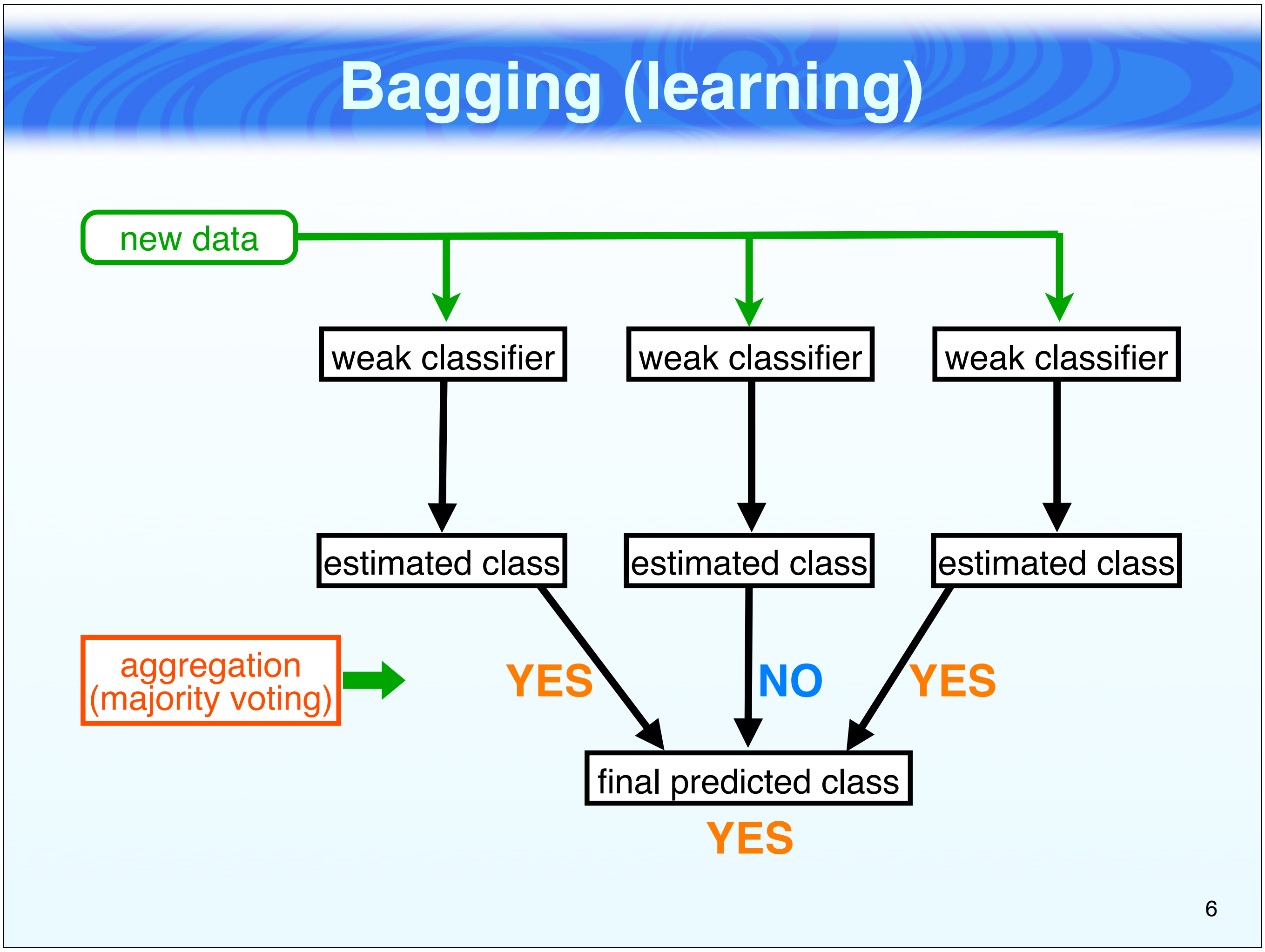
weak classifier

weak classifier

We begin with bagging, because our BaggTaming is a variant of a bagging. Briefly speaking, multiple weak classifiers are learned from bootstrapped training sets, and the predictions of these classifiers are aggregated.

In detail, multiple training data sets are first generated by bootstrap sampling from an original training data set.

Each training set is fed to a weak learner, and weak classifiers are learned. Any supervised learning methods, such as naive Bayes or SVM, can be used as a weak learner.



Bagging (learning)

new data

weak classifier

weak classifier

weak classifier

estimated class

estimated class

estimated class

aggregation (majority voting)

YES

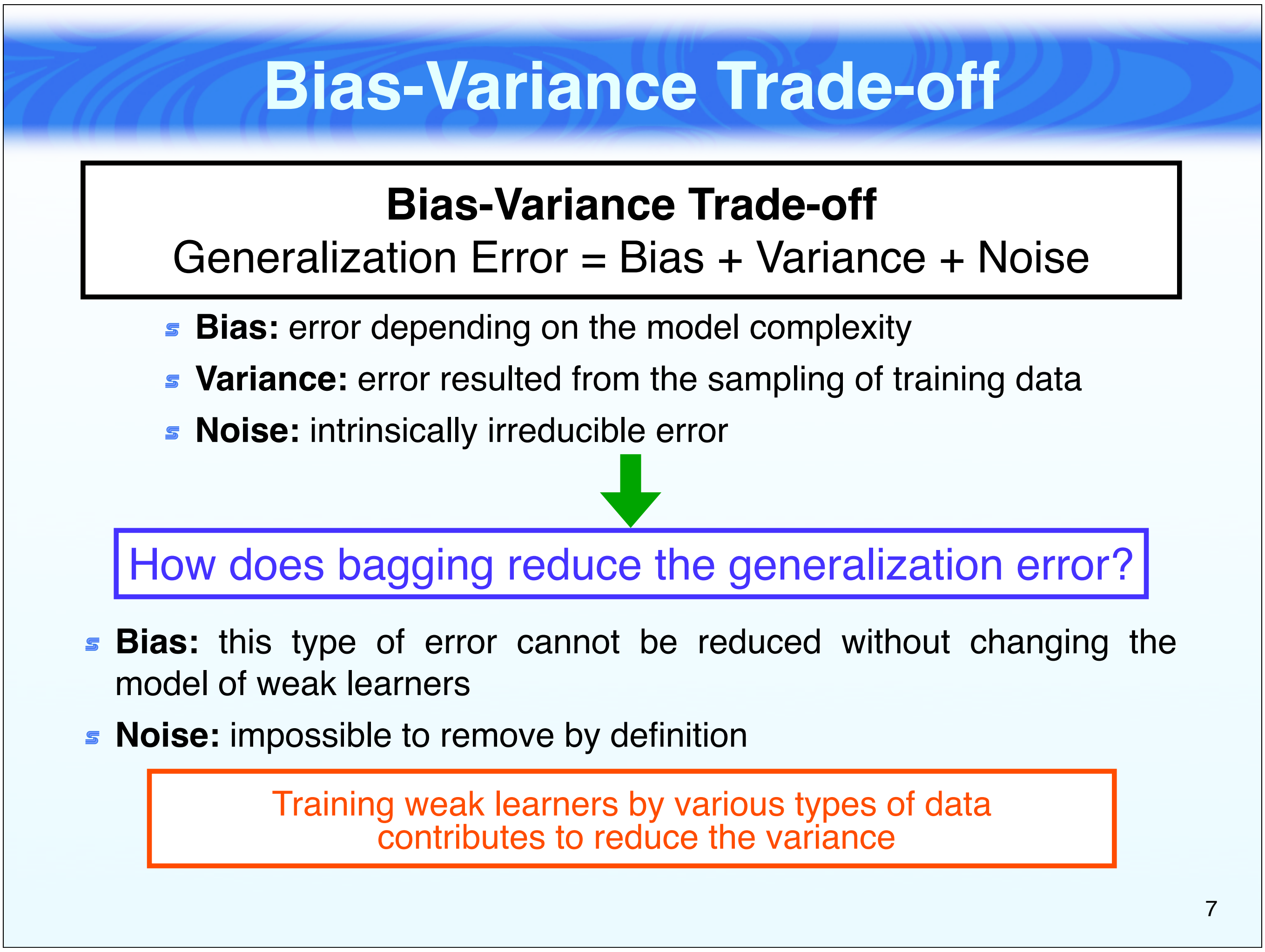
NO

YES

final predicted class

YES

Once weak classifiers are learned, the final class is predicted as follows.
New data to classify are fed to each weak classifier, and each classifier outputs its estimated class.
The final class is the majority class among these estimated classes.



Bias-Variance Trade-off

Bias-Variance Trade-off
Generalization Error = Bias + Variance + Noise

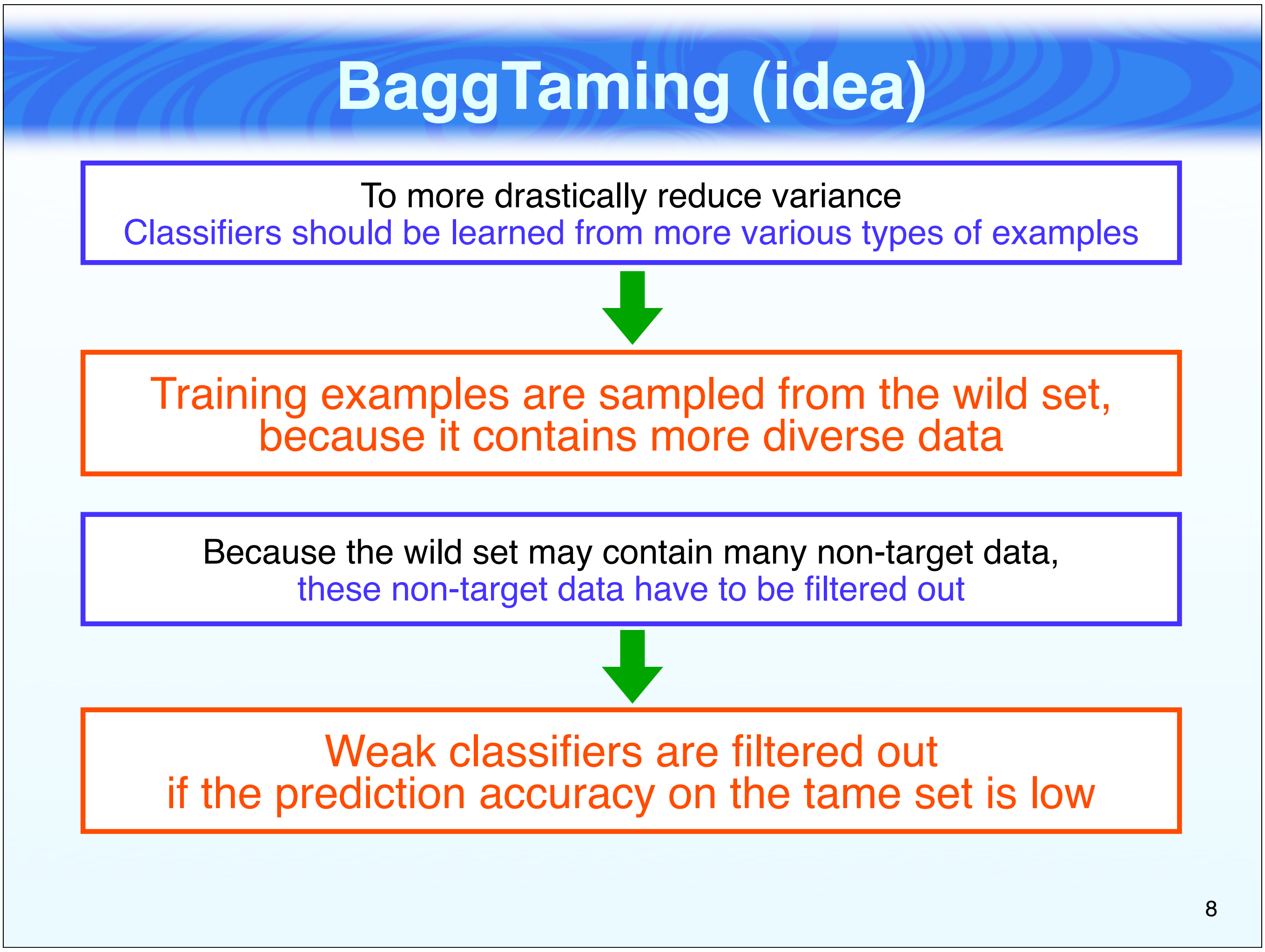
- **Bias:** error depending on the model complexity
- **Variance:** error resulted from the sampling of training data
- **Noise:** intrinsically irreducible error

How does bagging reduce the generalization error?

- **Bias:** this type of error cannot be reduced without changing the model of weak learners
- **Noise:** impossible to remove by definition

Training weak learners by various types of data contributes to reduce the variance

Briman showed the reason why the prediction accuracy is improved by bagging based on bias-variance trade-off.
The generalization error can be decomposed into three parts: bias, variance, and noise.
The bias is error depending on the model complexity.
The variance is error resulted from the sampling of training data, and the noise is intrinsically irreducible error.
Generally speaking, by reducing the model complexity, the bias can be decreased, but the variance is increased, and vice versa.
Bagging cannot reduce bias and noise because of these reasons, but training weak learners by various types of data contributes to reduce the variance.
In summary, bagging is a technique to reduce variance without sacrificing bias.



BaggTaming (idea)

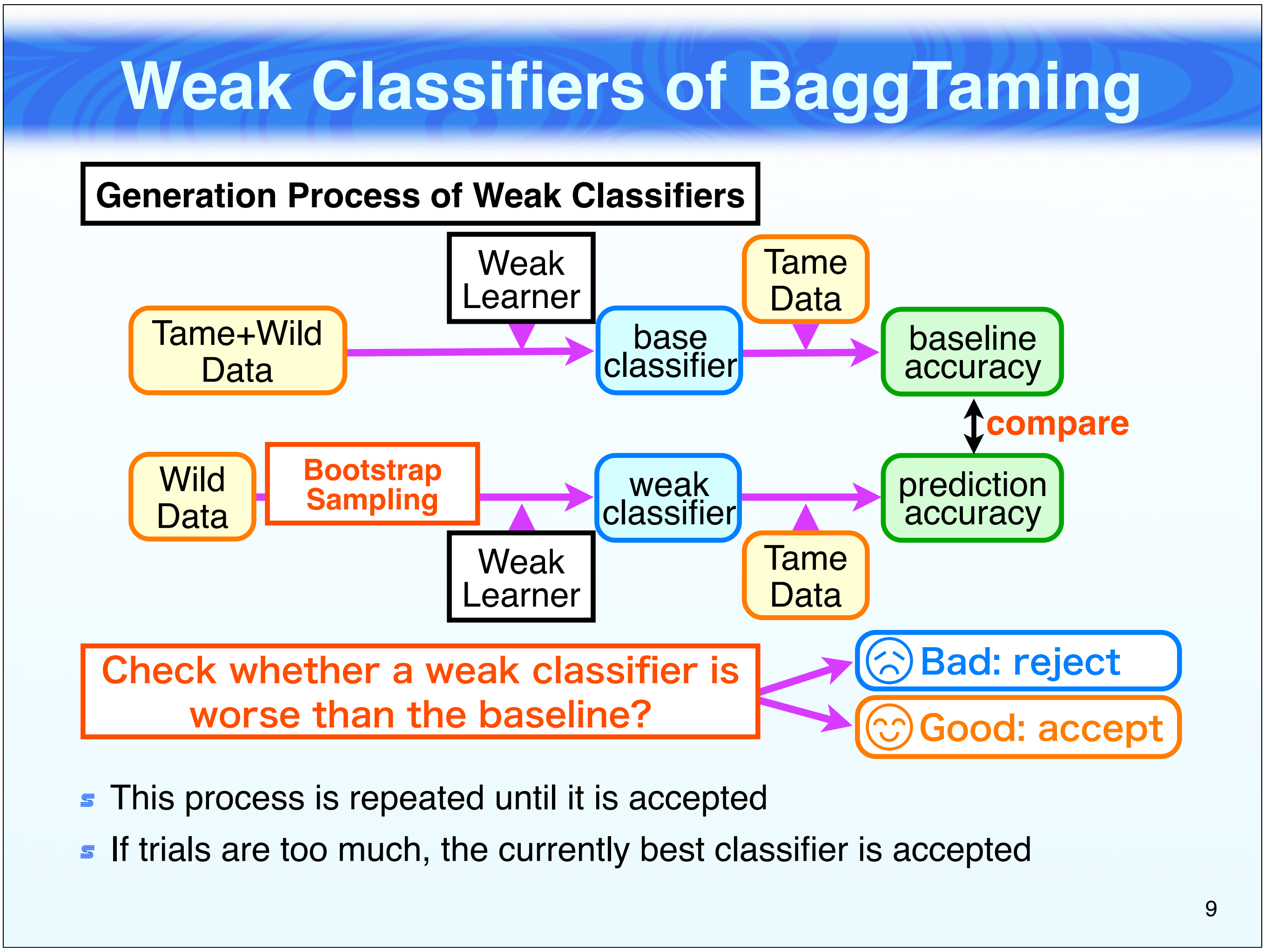
To more drastically reduce variance
Classifiers should be learned from more various types of examples

Training examples are sampled from the wild set,
because it contains more diverse data

Because the wild set may contain many non-target data,
these non-target data have to be filtered out

Weak classifiers are filtered out
if the prediction accuracy on the tame set is low

Based on the theory of bias-variance trade-off, we discuss the idea of our BaggTaming.
In order to more drastically reduce the variance, classifiers should be learned from more various types of examples. For this purpose, training examples are sampled from the wild set, because it contains more diverse data.
But, we now face one difficulty. Because the wild set may contain many non-target data, these data have to be filtered out.
For this purpose, we use a tame data set. Weak classifiers are filtered out if the prediction accuracy on the tame set is low.



Weak Classifiers of BaggTaming

Generation Process of Weak Classifiers

Tame+Wild Data

Weak Learner

base classifier

Tame Data

baseline accuracy

compare

Wild Data

Bootstrap Sampling

Weak Learner

weak classifier

Tame Data

prediction accuracy

Check whether a weak classifier is worse than the baseline?

Bad: reject

Good: accept

- This process is repeated until it is accepted
- If trials are too much, the currently best classifier is accepted

We show the detail of the generation process of weak classifiers of BaggTaming. Before learning weak classifiers, we compute the baseline prediction accuracy. Tame and wild data are merged, and weak learner acquires a base classifier from this merged data. The prediction accuracy on the tame set is considered as the baseline accuracy.

Next, weak classifiers are learned. Training examples are generated by bootstrap sampling from the wild data set. From these examples, a candidate weak classifier is acquired, and the prediction accuracy on the tame set is calculated.

The accuracy is compared with the baseline. If it is worse than the baseline the candidate weak classifier is rejected; otherwise, it is accepted. This process is repeated until it is accepted. To avoid the infinite loop, if the number of iteration exceeds the threshold, the currently best classifier is accepted by default.

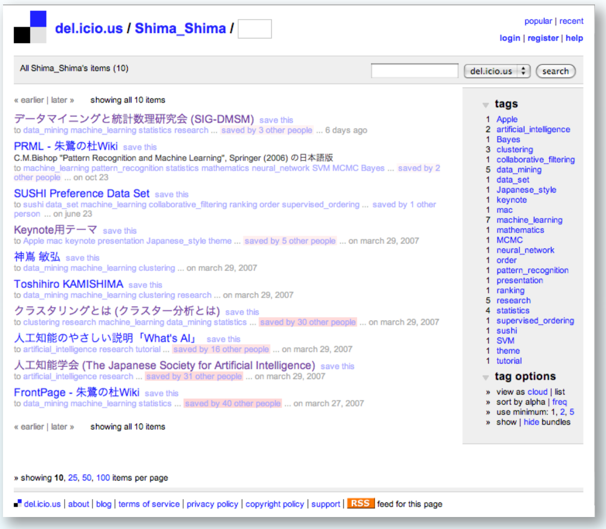
By repeating this process, multiple classifiers are generated. As in standard bagging, the final result of BaggTaming is derived by majority voting.

Next, we apply this BaggTaming to the tag personalization for a social bookmarking service.

Collaborative Tagging

Social Bookmark Service

- Users can **register their favorite Web Pages**
- To these Web pages, users can **assign tags** to attribute them
- These Web pages and tags can be **shared with other users**



del.icio.us / Shima_Shima /

popular | recent
login | register | help

All Shima_Shima's items (10)

del.icio.us search

« earlier | later » showing all 10 items

データマイニングと統計数理研究会 (SIG-DMSM) save this
to data_mining machine_learning statistics research ... saved by 3 other people ... 6 days ago

PRML - 朱鷺の杜Wiki save this
C.M.Bishop "Pattern Recognition and Machine Learning", Springer (2006) の日本語版
to machine_learning pattern_recognition statistics mathematics neural_network SVM MCMC Bayes ... saved by 2 other people ... on oct 23

SUSHI Preference Data Set save this
to sushi data_set machine_learning collaborative_filtering ranking order supervised_ordering ... saved by 1 other person ... on june 23

Keynote用テーマ save this
to Apple mac keynote presentation Japanese_style theme ... saved by 5 other people ... on march 29, 2007

神嶌 敏弘 save this
to data_mining machine_learning clustering ... on march 29, 2007

Toshihiro KAMISHIMA save this
to data_mining machine_learning clustering research ... on march 29, 2007

クラスタリングとは (クラスター分析とは) save this
to clustering research machine_learning data_mining statistics ... saved by 30 other people ... on march 29, 2007

人工知能のやさしい説明「What's AI」 save this
to artificial_intelligence research tutorial ... saved by 16 other people ... on march 29, 2007

人工知能学会 (The Japanese Society for Artificial Intelligence) save this
to artificial_intelligence research ... saved by 31 other people ... on march 29, 2007

FrontPage - 朱鷺の杜Wiki save this
to data_mining machine_learning statistics ... saved by 40 other people ... on march 27, 2007

« earlier | later » showing all 10 items

tags

1	Apple
2	artificial_intelligence
1	Bayes
3	clustering
1	collaborative_filtering
5	data_mining
1	data_set
1	Japanese_style
1	keynote
1	mac
7	machine_learning
1	mathematics
1	MCMC
1	neural_network
1	order
1	pattern_recognition
1	presentation
1	ranking
5	research
4	statistics
1	supervised_ordering
1	sushi
1	SVM
1	theme
1	tutorial

tag options

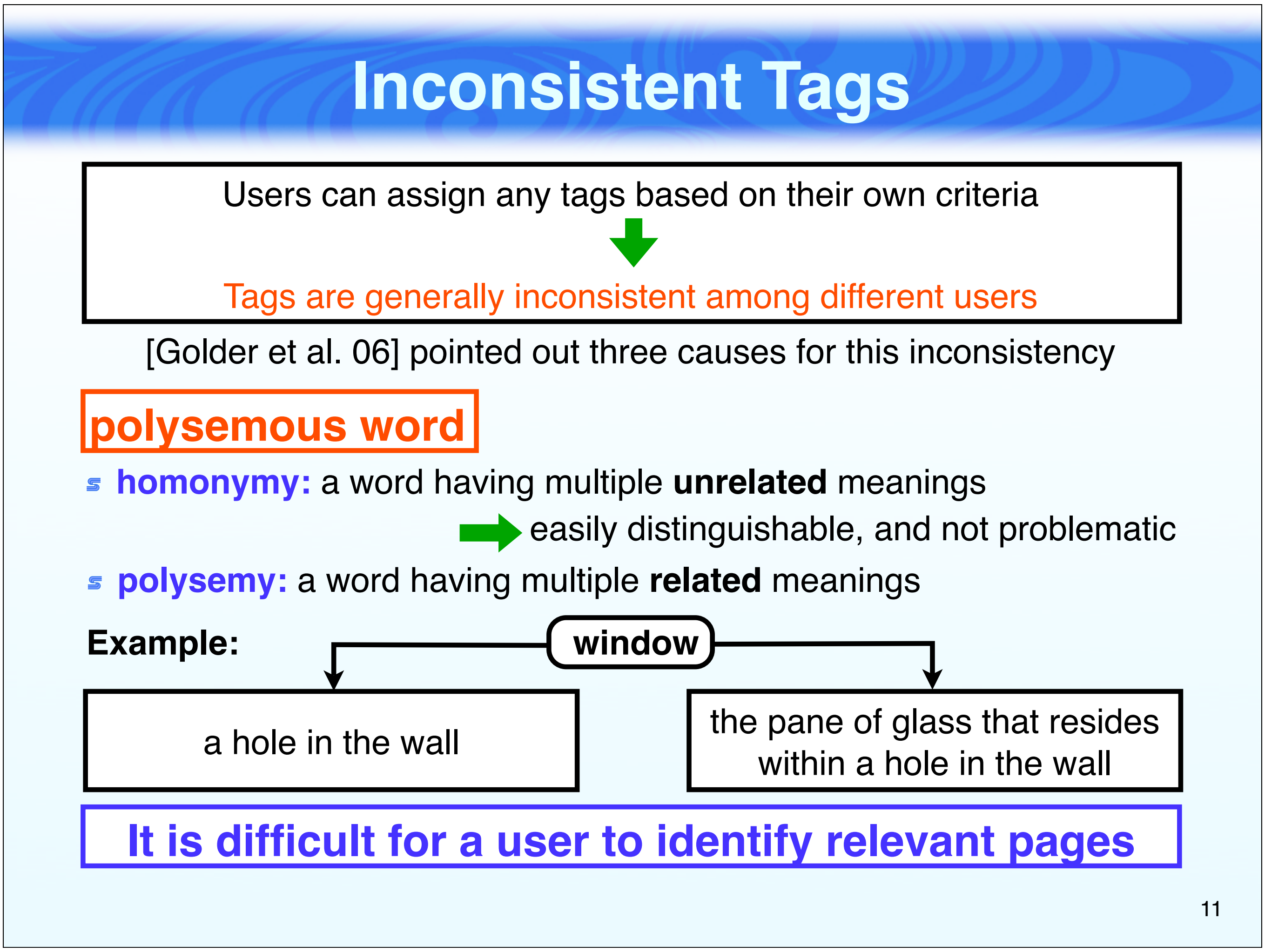
» view as cloud | list
» sort by alpha | freq
» use minimum: 1, 2, 5
» show | hide bundles

» showing **10**, 25, 50, 100 items per page

del.icio.us | about | blog | terms of service | privacy policy | copyright policy | support | RSS feed for this page

Shared tags can be exploited
for classifying or retrieving Web pages

We first talk about collaborative tagging, such as a social bookmarking.
In this service, users can register their favorite Web pages. To these Web pages, users can assign to attribute them. These Web pages and tags can be shared with other users.
These shared tags can be exploited for classifying or retrieving Web pages.



Inconsistent Tags

Users can assign any tags based on their own criteria

Tags are generally inconsistent among different users

[Golder et al. 06] pointed out three causes for this inconsistency

polysemous word

- **homonymy:** a word having multiple **unrelated** meanings
  ➡ easily distinguishable, and not problematic
- **polysemy:** a word having multiple **related** meanings

Example:

window

a hole in the wall

the pane of glass that resides within a hole in the wall

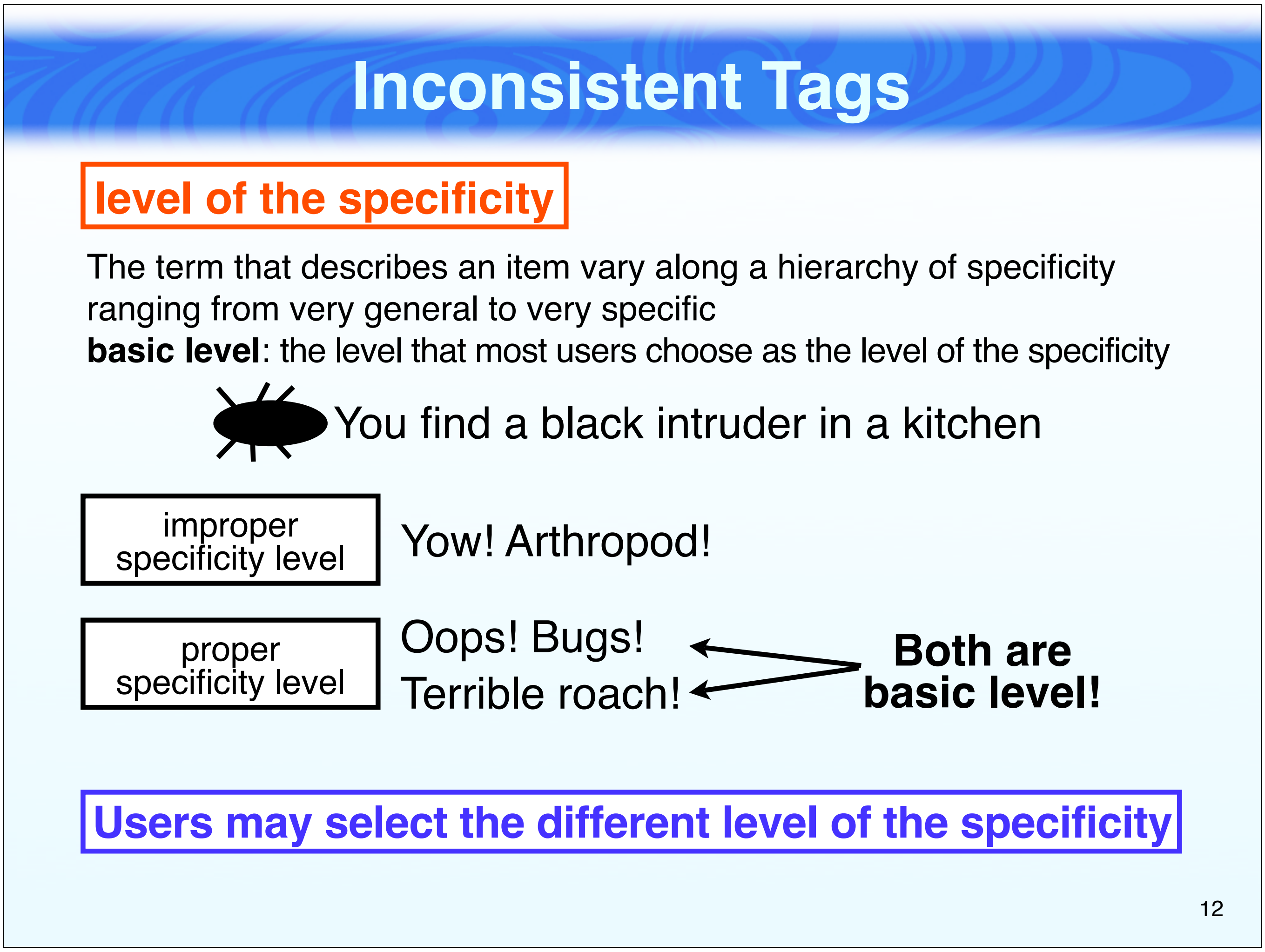
It is difficult for a user to identify relevant pages

In collaborative tagging, users can assign any tags freely. Therefore, tags are generally inconsistent among users. Golder et al. pointed out three causes for this inconsistency.

The first one is a polysemous word.

Polysemy refers a word having multiple related meanings. For example, someone use a word “window” to refer a hole in the wall. Another refers the pane of glass that resides within the hole.

Because the related meanings confuse users, it is difficult for users to find relevant pages.



Inconsistent Tags

level of the specificity

The term that describes an item vary along a hierarchy of specificity ranging from very general to very specific

basic level: the level that most users choose as the level of the specificity

You find a black intruder in a kitchen

improper specificity level	Yow! Arthropod!
proper specificity level	Oops! Bugs! Terrible roach!

Both are basic level!

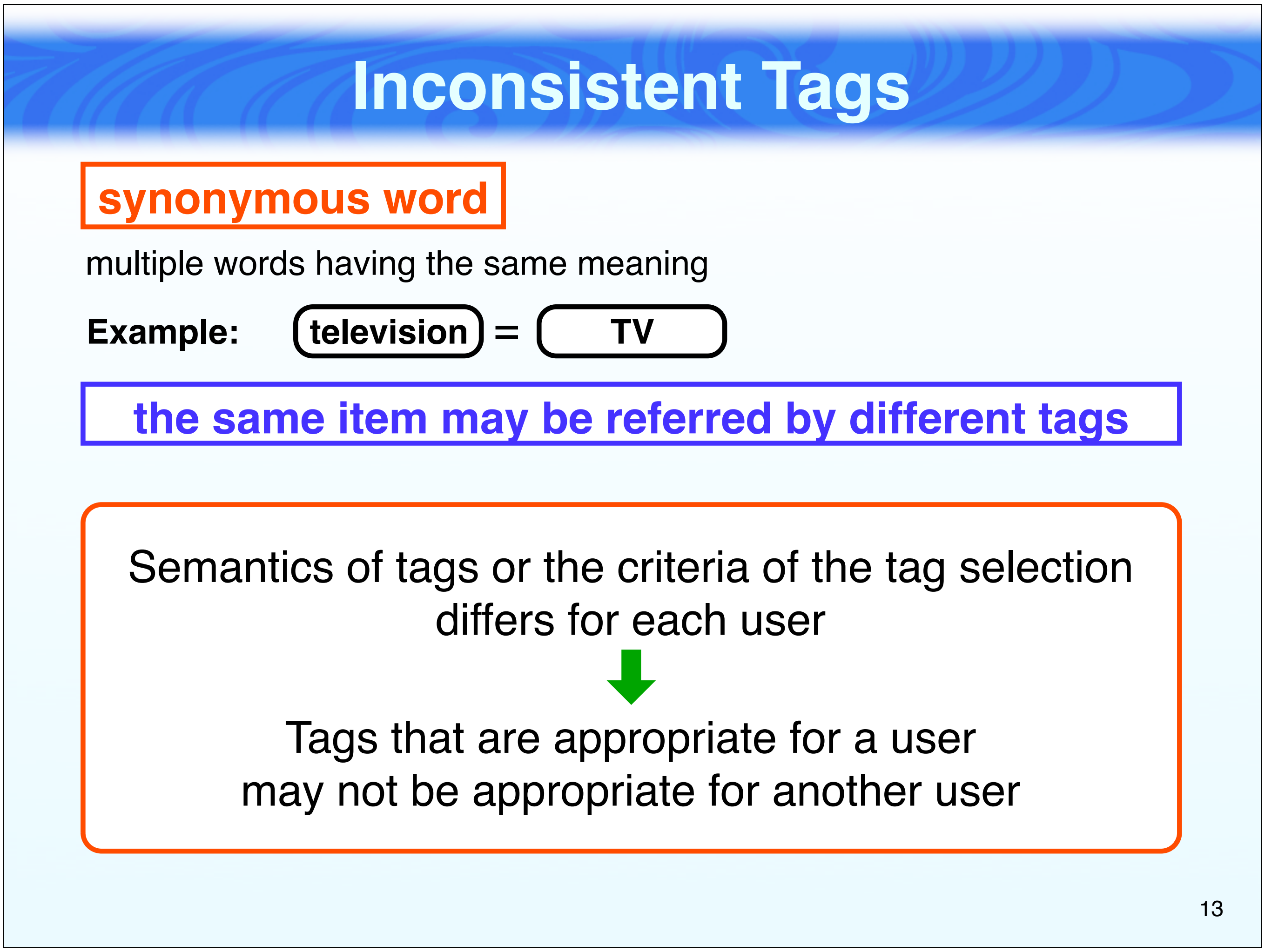
Users may select the different level of the specificity

The second cause is the level of the specificity.

The term that describes an item vary along a hierarchy of specificity ranging from very general to very specific. Most users choose the basic level as the specificity level. But, the basic level may not be unique.

For example, you find a black intruder in a kitchen. No one screams “Yow! Arthropod!” However, one would say “Oops! Bugs!”, while another would say “Terrible roach!” Both of them can be considered as the basic level.

Users may select the different level of the specificity, and different tags can be assigned to the same page.



Inconsistent Tags

synonymous word

multiple words having the same meaning

Example: **television** = **TV**

the same item may be referred by different tags

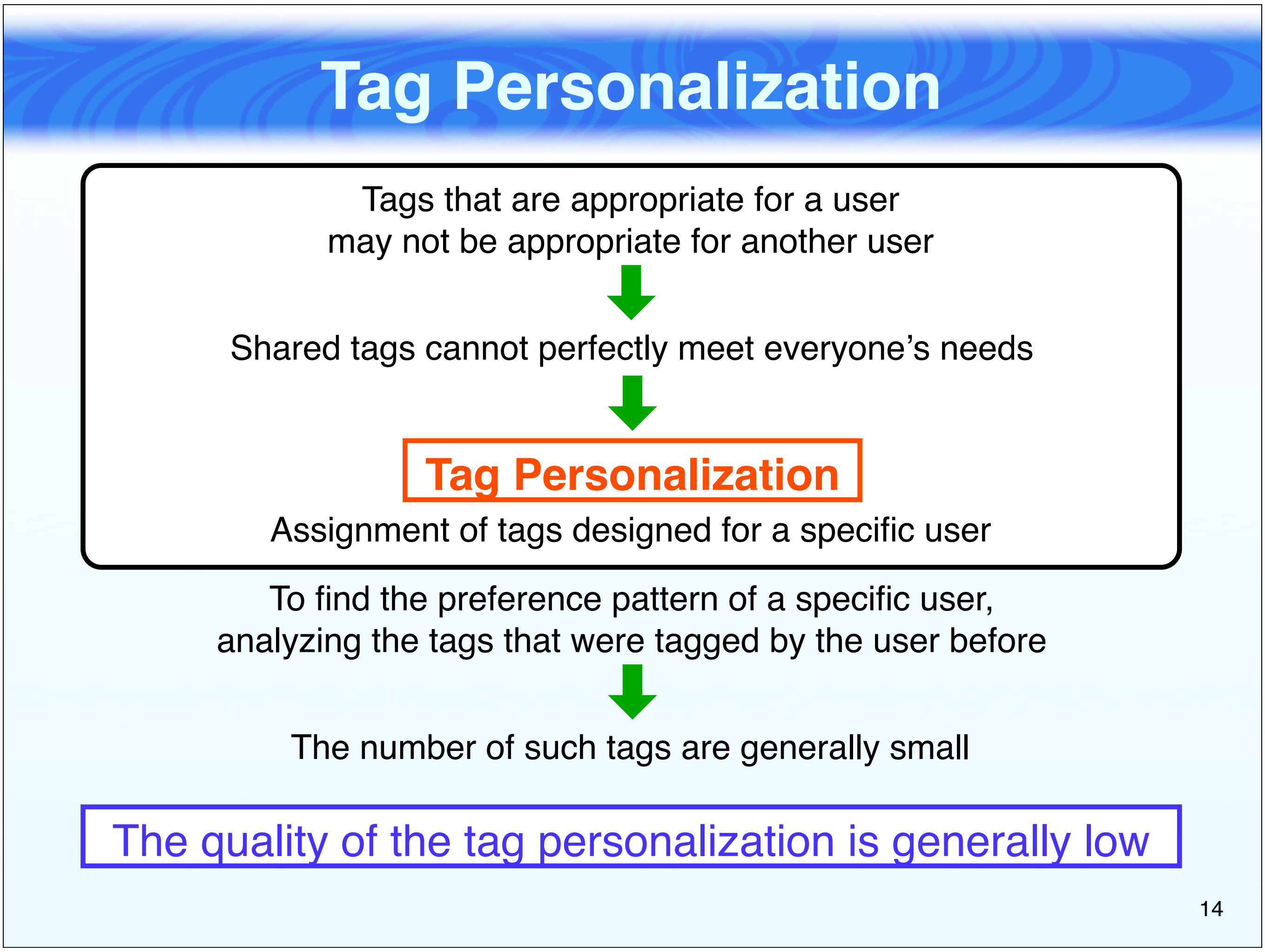
Semantics of tags or the criteria of the tag selection differs for each user

⬇

Tags that are appropriate for a user may not be appropriate for another user

The final cause is synonymous words. Synonymy refers that the multiple words having the same meaning. For example, television and TV. In this case, the same item may be referred by different tags.

Semantic of tags or the criteria of the tag selection differs for each user; thus, tags that are appropriate for a user may not be appropriate for another.



Tag Personalization

Tags that are appropriate for a user
may not be appropriate for another user

Shared tags cannot perfectly meet everyone’s needs

Tag Personalization

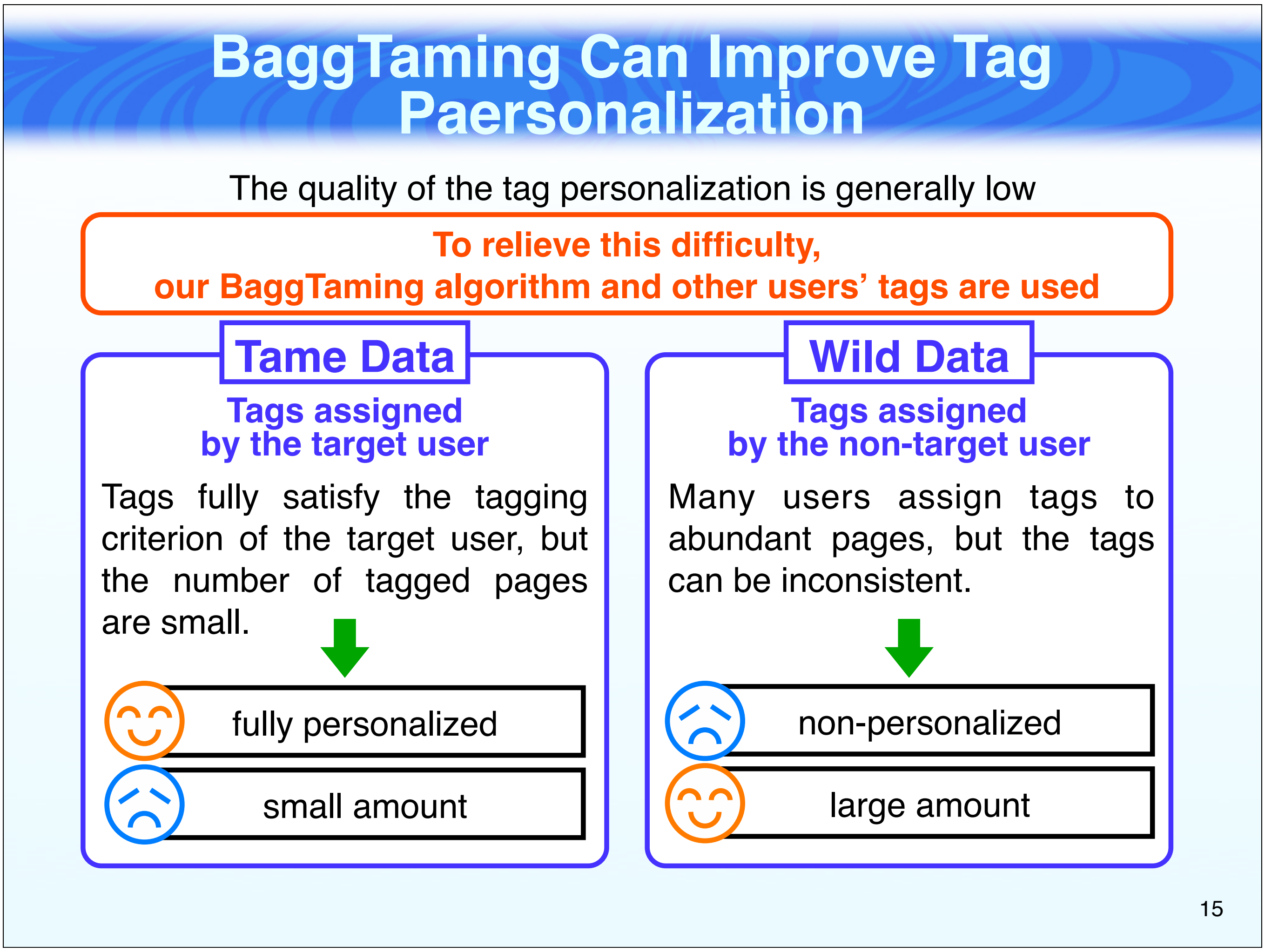
Assignment of tags designed for a specific user

To find the preference pattern of a specific user,
analyzing the tags that were tagged by the user before

The number of such tags are generally small

The quality of the tag personalization is generally low

In such a case, shared tags cannot perfectly meet everyone’s needs. So, we try the tag personalization, that is the assignment of tags designed for a specific user.
To find the preference pattern of a specific user, all that have to do is analyzing the tags that were tagged by the user before. However, the number of such tags are generally small. Due to this shortage of training examples, the quality of the tag personalization is generally low. In a context of recommendation, this problem is called by “a cold-start problem.”



BaggTaming Can Improve Tag Paersonalization

The quality of the tag personalization is generally low

To relieve this difficulty,
our BaggTaming algorithm and other users' tags are used

Tame Data

Tags assigned by the target user

Tags fully satisfy the tagging criterion of the target user, but the number of tagged pages are small.

☺ fully personalized

☹ small amount

Wild Data

Tags assigned by the non-target user

Many users assign tags to abundant pages, but the tags can be inconsistent.

☹ non-personalized

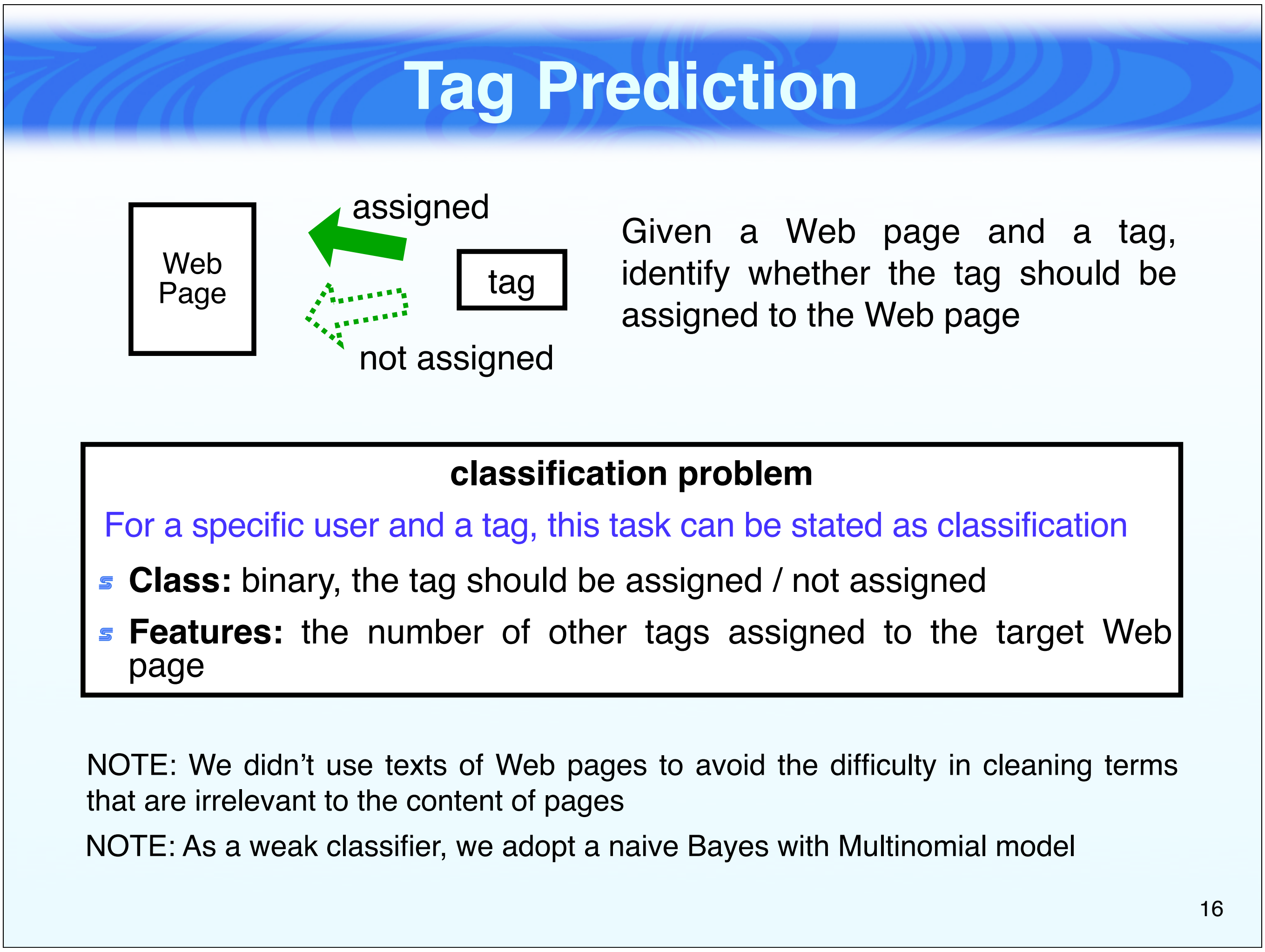
☺ large amount

To relieve this difficulty, we use our BaggTaming algorithm and other users' tags.

Tags assigned by the target user are considered as tame data. These data are fully personalized, but the number of data is relatively small.

On the other hand, tags assigned by the non-target user are considered as wild data. These not fully personalized, but are much more abundant.

Employing our BaggTaming would improve the quality of the tag personalization. Next, we empirically show this improvement.



Tag Prediction

Web Page

assigned

tag

not assigned

Given a Web page and a tag, identify whether the tag should be assigned to the Web page

classification problem

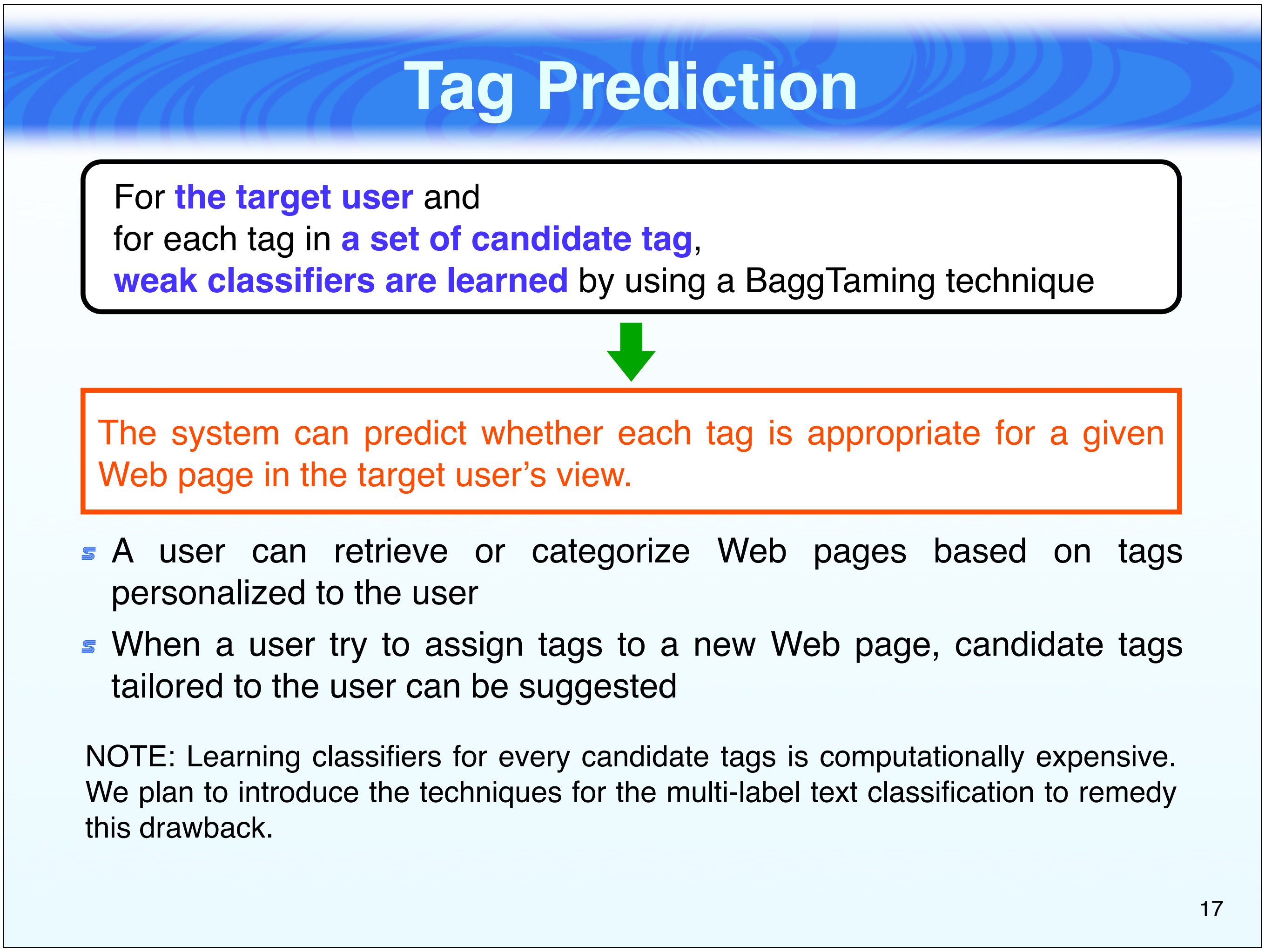
For a specific user and a tag, this task can be stated as classification

- **Class:** binary, the tag should be assigned / not assigned
- **Features:** the number of other tags assigned to the target Web page

NOTE: We didn’t use texts of Web pages to avoid the difficulty in cleaning terms that are irrelevant to the content of pages

NOTE: As a weak classifier, we adopt a naive Bayes with Multinomial model

The personalized tag prediction task is formalized as classification. Specifically, a class is binary, assigned or not-assigned. As features, we adopted the number of other tags assigned to the target Web page.



Tag Prediction

For **the target user** and
for each tag in **a set of candidate tag**,
weak classifiers are learned by using a BaggTaming technique

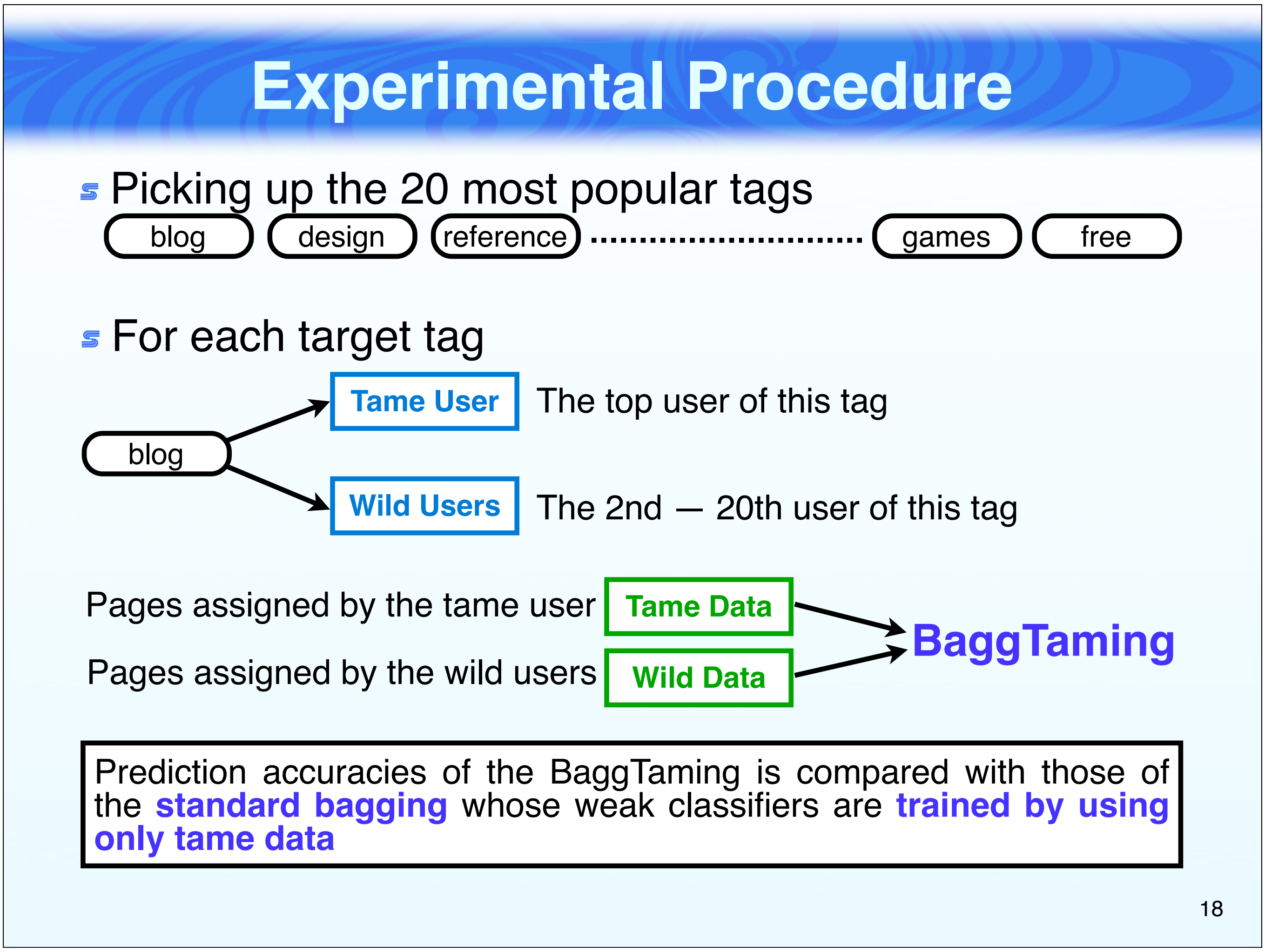
↓

The system can predict whether each tag is appropriate for a given Web page in the target user’s view.

- A user can retrieve or categorize Web pages based on tags personalized to the user
- When a user try to assign tags to a new Web page, candidate tags tailored to the user can be suggested

NOTE: Learning classifiers for every candidate tags is computationally expensive. We plan to introduce the techniques for the multi-label text classification to remedy this drawback.

For the target user and for each tag in a set of candidate tag, weak classifiers are learned by using a BaggTaming technique. Once classifiers are learned, the system can predict whether each tag is appropriate for a given Web page in the target user’s view.
This personalized tag prediction used for the purpose like these.



Experimental Procedure

- Picking up the 20 most popular tags

- For each target tag

Tame User: The top user of this tag

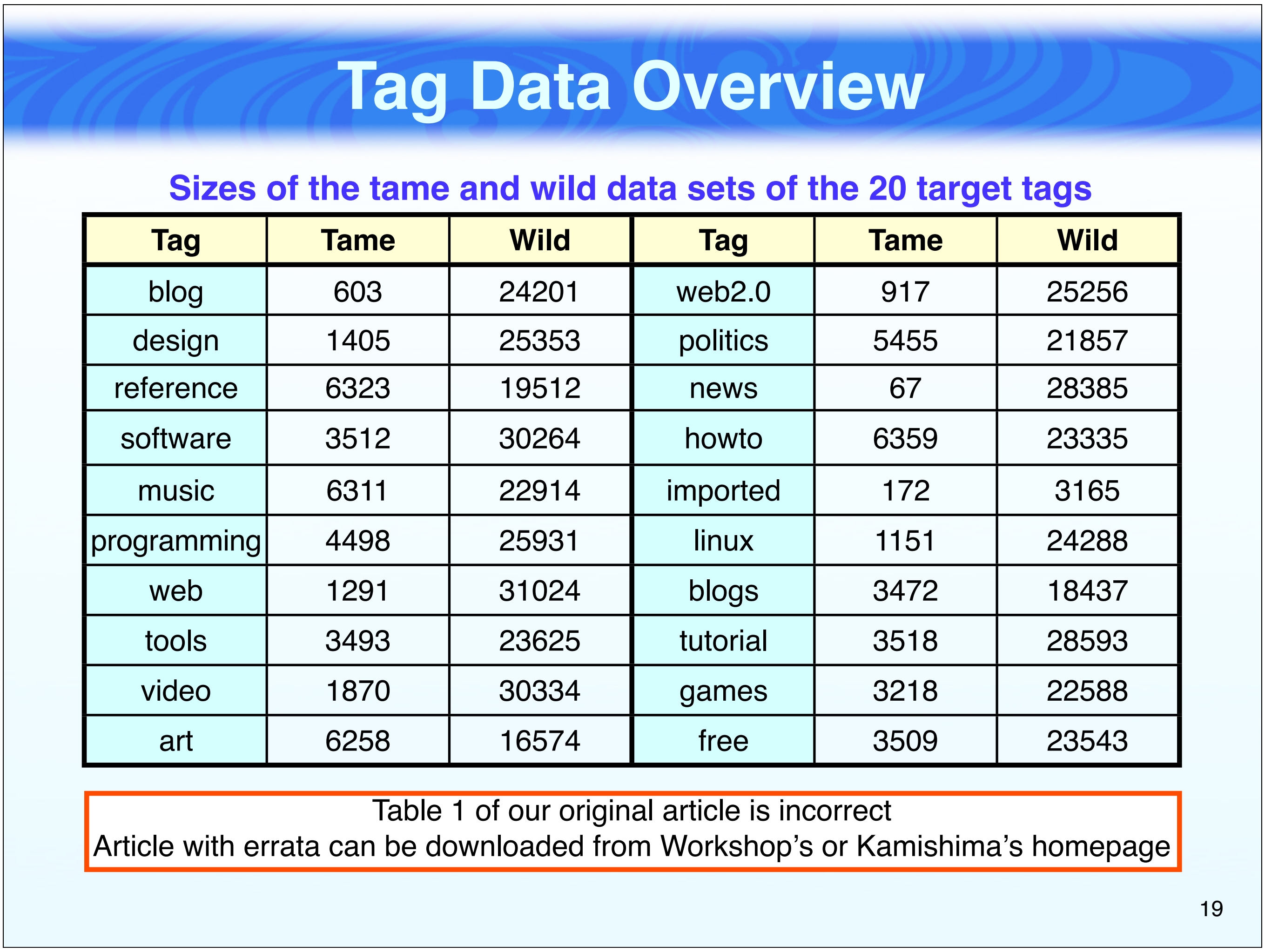
Wild Users: The 2nd — 20th user of this tag

Pages assigned by the tame user → Tame Data

Pages assigned by the wild users → Wild Data

Prediction accuracies of the BaggTaming is compared with those of the **standard bagging** whose weak classifiers are **trained by using only tame data**

We next show an experimental procedure.
As the target tags, we picked up the 20 most popular tags.
For each target tag, the top user of this tag is a tame user, and the second to twentieth users are wild users.
Pages assigned by the tame user and the wild users are treated as tame and wild data, respectively.
Prediction accuracies of the BaggTaming is compared with those of the standard bagging whose weak classifiers are trained by using only tame data.



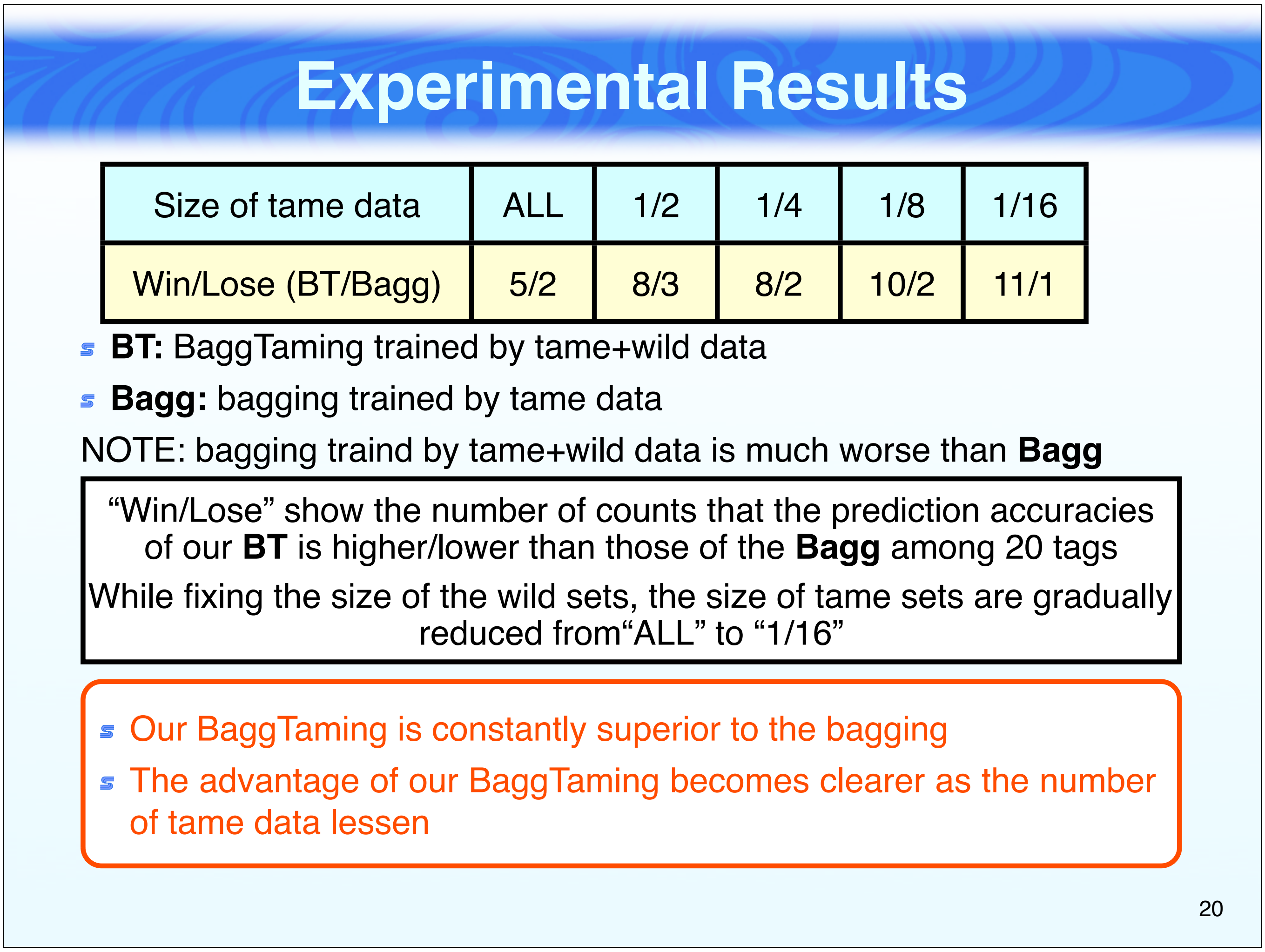
Tag Data Overview

Sizes of the tame and wild data sets of the 20 target tags

Tag	Tame	Wild	Tag	Tame	Wild
blog	603	24201	web2.0	917	25256
design	1405	25353	politics	5455	21857
reference	6323	19512	news	67	28385
software	3512	30264	howto	6359	23335
music	6311	22914	imported	172	3165
programming	4498	25931	linux	1151	24288
web	1291	31024	blogs	3472	18437
tools	3493	23625	tutorial	3518	28593
video	1870	30334	games	3218	22588
art	6258	16574	free	3509	23543

Table 1 of our original article is incorrect
Article with errata can be downloaded from Workshop’s or Kamishima’s homepage

This show the sizes of the tame and wild data sets of the 20 target tags.
You can see that the number of the tame data is much smaller than that of the wild data.
Here, we apologize that Table 1 of our original article incorrect. Article with errata can be downloaded from Workshop’s or Kamishima’s homepage.



Experimental Results

Size of tame data	ALL	1/2	1/4	1/8	1/16
Win/Lose (BT/Bagg)	5/2	8/3	8/2	10/2	11/1

- **BT:** BaggTaming trained by tame+wild data
- **Bagg:** bagging trained by tame data

NOTE: bagging traind by tame+wild data is much worse than **Bagg**

"Win/Lose" show the number of counts that the prediction accuracies of our **BT** is higher/lower than those of the **Bagg** among 20 tags

While fixing the size of the wild sets, the size of tame sets are gradually reduced from"ALL" to "1/16"

- Our BaggTaming is constantly superior to the bagging
- The advantage of our BaggTaming becomes clearer as the number of tame data lessen

This is a summary of experimental results. The prediction accuracies on the tame data sets are compared.

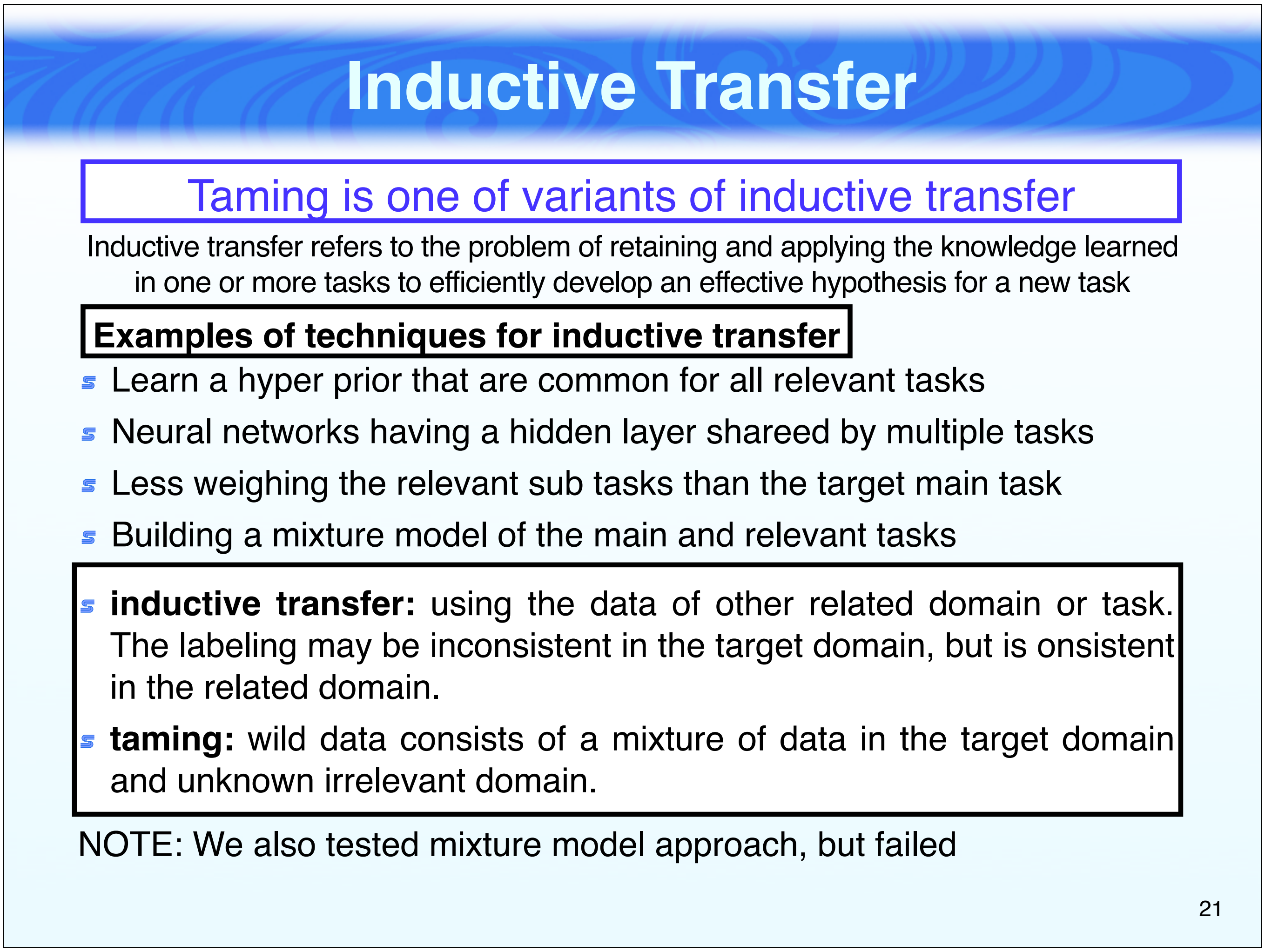
BT and Bagg mean results of our BaggTaming and a standard bagging. We show the number of tags that the our BaggTaming wins or loses among 20 target tags. For example, "5/2" means that our BaggTaming is significantly ruperior to a standard bagging in five tags, and is significantly inferior in two tags.

Further, we also tested the case where the number of tame data is much smaller. For example, the "1/2" column shows the results when the number of tame data is reduced to a half of the original while fixing the size of the wild sets.

It would be reasonable to say that these two conclusions:

Our BaggTaming is constantly superior to the bagging.

The advantage of our BaggTaming becomes clearer as the number of tame data lessen. BaggTaming is useful when the tame data is less available. This observation enhance the usefulness of our BaggTaming.



Inductive Transfer

Taming is one of variants of inductive transfer

Inductive transfer refers to the problem of retaining and applying the knowledge learned in one or more tasks to efficiently develop an effective hypothesis for a new task

Examples of techniques for inductive transfer

- Learn a hyper prior that are common for all relevant tasks
- Neural networks having a hidden layer shareed by multiple tasks
- Less weighing the relevant sub tasks than the target main task
- Building a mixture model of the main and relevant tasks

- **inductive transfer:** using the data of other related domain or task. The labeling may be inconsistent in the target domain, but is onsistent in the related domain.
- **taming:** wild data consists of a mixture of data in the target domain and unknown irrelevant domain.

NOTE: We also tested mixture model approach, but failed

We briefly discuss related work. Taming is one of variants of inductive transfer or transfer learning.
Inductive transfer is not formally defined but refers the problem of retaining and applying the knowledge learned in one or more tasks to efficiently develop an effective hypothesis for a new task.
In our opinion, our taming differs from inductive transfer in this point.
Inductive transfer uses the data of other related domain or task. The labeling may be inconsistent in the target domain, but is consistent in the related domain.
In taming, wild data consists of a mixture of data for the target and unknown irrelevant tasks.

Conclusion

Summary

- **Stating the taming approach**
  Prediction accuracy was improved by using a small set of reliable tame data together with less reliable abundant wild data
- **Developing BaggTaming algorithm**
  Ensemble learning sampling from wild data, and weak classifiers are filtered out by exploiting tame data
- **Application to collaborative tagging data**
  Personalized tags are more precisely predicted by adopting our BaggTaming technique

Homepage

- http://www.kamishima.net/ (errata of Table 1 can be downloaded)

Future Work

- **Using formal ontology**
  Using the labels in formal ontology will realize highly consistent tags, but it is difficult to label so many documents.
  By adopting a taming approach together with collaboratively tagged documents, one can classify much more documents by using vocabulary of a formal ontology.
- **Improvement of efficiency**
  Our current sampling technique is highly brute forced and inefficient.
  Adaptive sampling will contribute to alleviate this inefficiency.
- **Multi-label technique**
  Constructing classifiers for every tags is computationally expensive.
  A multi-label classification technique would be useful to remove this drawback.