Abstract: As most of us spend a substantial amount of time on computers nowadays for either work or personal tasks, it is very easy to realise that interruptions occur on a regular basis. Such interruptions often tend to cause an information and cognitive overload due to the inundation of data entering our desktop which disrupt us from our current task. A few years back, people used to be able to interrupt you only by calling or physically walking into an office for example. Nowadays, through various technological advancements, interruptions can occur via e-mails, instant messaging, mobiles and countless other mechanisms. In this paper we present DIME, a solution to interruption management to reduce disruptive interruptions by understanding the user's desktop context and current activities, and evaluated based on e-mail communications. Furthermore, our initial evaluation methodology proved it to be very promising.

1 Introduction and Background

Interruption is indeed a major headache in today's workplace most particularly for knowledge workers (KWs). The issue is that as society has moved to a technological era this has created new challenges in the aspect of dealing with various interrupts where in most cases, these are disruptive from the current work one was attending to.

[MGH05] observed 24 KWs during their work and showed that work tends to be highly fragmented. Work fragmentation is "a break in continuous work activity" [MGH05] which is when a short time span is spent on a task and frequent switching occurs. Work fragmentation has increasingly become a way of life where people spend around 11 minutes working before switching to another task. Furthermore, although task switching is sometimes valuable since having a short break tends to facilitate task performance, frequent task switching between a large number of tasks could be detrimental [MGH05].

We focus on the area of interruption management on one's personal desktop and aim to find a solution to reduce disruptive interruptions by understanding the user's desktop context and current activities. Through our approach it is possible to decide if the user should receive an interrupting communication (e.g. e-mail message) immediately or at a more opportune time.
2 Interruption Manager

DIME is based on machine learning, in particular Naïve Bayesian Classifiers. We automatically obtain information regarding the user's desktop activities as she is working. After a period of training, the system is able to recognise activities on the user's desktop which are of a non-interruptable nature (i.e. user is in a busy state) and activities during which the user is available. We also present an environment to aid the user in reducing interruptions and decrease the resumption lag [AT04] once the interruption has been attended to.

The system uses a set of states based on a traffic light convention of Red-Amber-Green. Red identifies that the user is busy and cannot be disturbed (apart from communications which are deemed to be important and contacts which are on a priority list and can still interrupt the user). In an Amber state, the system is undergoing training for busy activities and Green is a state of interruptibility. The main goals are:

1. Learn the interruptible and non-interruptive state the user is in
2. Minimise user input.
3. Help decrease the disruptive effects of interruptions: Interruptions bring about stress, reduce an individual's performance and generally make a person feel frustrated and sometimes helpless [AB04]. Initially we focus on e-mail interruptions.
4. Provide an interface which does not cause cognitive fatigue
5. Provide an Extendible Environment: We cater for future developers to create application wrappers to interact with the system. Extensibility is also approached from the aspect of semantically sharing information through the use of RDF.

2.2 System Overview

Machine Learning and User Modelling: DIME identifies the user's state (busy or available) by applying a machine learning technique based on the probabilistic nature of Bayesian Classifiers (similar to that presented in [HA03 and Ho07]) to infer a value based on the prior training received. Once training has initiated mouse, keyboard and application monitoring procedures take place. We also monitor the user browsing for websites flagged as interruptable (to automatically change status to available). The list of websites is stored in RDF format and it's semantic graph is created through OpenCalais1. Moreover, if the user has pre-defined a moment as being "busy" through calendar events, then the status is changed and no classification of current activities will take place till the event has terminated.

User Centric Design: The more usable a program is, the less likely a user feels stressed, helpless and subjected to cognitive overload. We designed DIME to be easy to use and require minimal user interaction. We kept in focus [Ho99] suggestions for interfaces, which consider issues such as timing services based on user's attentional focus. Taking

11A free semantic web service initiated by Thomson Reuters http://www.opencalais.com/.
[MCB03]'s advice, our notifications are implemented in a way to convey a short amount of information to the user and do not require prolonged attention.

*Delivering communications which are relevant and their timing:* By understanding what potential interruptions are about, we identify if these are of high priority by considering the contents. We make use of semantic analysis (through OpenCalais) as well as string similarity algorithms to identify topics and people that require urgent attention. We also use a set of topics and keywords which typically put communications in high priority. The user may also add people on a priority list for automated high priority. Such contacts are also stored in RDF to facilitate Semantic sharing of information. Additionally, to deliver communications at an opportune time (while busy), when such communications are deemed to be of high priority, when the user's window focus is changed this is considered to be a *presumed best* opportunity [AB04]. The reason is that when we change windows, we usually make some subconscious notes of what we need to continue working on (based on [Ze27]). In this way, the *resumption lag* [AT04] from the interruption is reduced, minimising the disruption effect.

### 3 Evaluation

For the usability aspect of DIME, Heuristic Evaluation [Ni92] was used. With the aspect of evaluating the learning capabilities of the system unfortunately there does not seem to be any standard technique to evaluation. A possible approach would be to identifying that ultimately the system became useful to the user. Total coverage of System Evaluation shall be assured by User Testing.

[Ni92] suggest that the accepted number of participants for a usability evaluation should be in the region of 3 to 5 participants.

We adhered to these suggestions and 4 participants were found, although they were not experts in usability studies. They used the system for 3-5 days and we obtained useful feedback about the system.

#### 3.1 Results

The feedback obtained by the participants was found to be very positive. No critical issues in the usability of the system were found. The most notable problem was the response time of the system in certain situations (such as displaying an email). The following feedback emerged:

1. All participants found it either easy or very easy to change between states.
2. All found the notification to be minimally intrusive.
3. With the overall functionality of the system, the users were mostly satisfied.
4. In terms of performance, one participant felt neutral about it whilst other participants found it to be mostly satisfying.
5. The accuracy of identifying busy moments was deemed to be mostly accurate.
6. Identification of opportune time windows was found to be neutral by one participant and mostly opportune by the rest.
7. Timeliness in emails being displayed was found to be mostly timely. Moreover it was agreed by all participants that the RAG representation of user states was appropriate. The system also became useful in a short time span between 2-4 hours.

4 Future Work and Conclusions

In the future, we recommend evaluating with more participants and suggest that the system is used during the most productive hours of the day. Future work may also focus on assigning priorities to communications by using a technique such as Support Vector Machines. The priority contact list could also be improved where users who have the program installed could view the current status of other contacts in their network. In this way there is an aspect of social judgment to minimise the interruptions.

The initial evaluation of DIME’s learning capability and its performance was positive and this is very encouraging. We plan to further this work and perform a more elaborate evaluation so that the system can be used in the real world, to reduce significantly the user's interruptions as while at work.

5 References


