<index-i> chỉ mục của từ thứ i.

<value-i> tần số xuất hiện của từ i.

Vì mỗi bình luận trên Twitter chỉ giới hạn 140 kí tự, nên số lượng các từ xuất hiện rất ít, trung bình từ 5 – 7 từ khi chưa xử lý wordnet, 10 - 12 từ khi đã xử lý wordnet. Nếu phải lưu tất cả các tần số xuất hiện của từng từ trong tweets, dữ liệu sẽ trở nên rất thưa, đa số đều mang giá trị 0, dẫn đến sự lãng phí bộ nhớ.

Nếu lưu trữ tiết kiệm bộ nhớ, trung bình sẽ có 6 đặc trưng tần số xuất hiện lớn hơn 0:

2:1 5:1 101:1 609:1 1200:1 15356:1

Với mỗi đặc trưng có dạng **index-i:value-i**, ta tốn khoảng 5 bytes, vậy 1 tweet chỉ tốn trung bình 30 bytes. 15.000 tweet sẽ chiếm 450.000 bytes. So với cách lưu trữ ban đầu, chúng ta tiết kiệm được 599.550.000 bytes.

Tất nhiên 15.000 tweets chỉ là một con số vô cùng nhỏ so với lượng dữ liệu trên Twitter. Nếu dữ liệu càng lớn, ý nghĩa của việc lưu trữ tiết kiệm bộ nhớ sẽ được thể hiện càng rõ.

2.3 Phân loại ý kiến bằng giải thuật máy học MNB

Multinomial Naïve Bayes (MNB) là một mô hình đơn giản nhưng hoạt động rất tốt trong việc phân loại văn bản. [Lewis & Gale, 1994] đã đề xuất kết hợp mô hình túi từ và NB tạo ra giải thuật Multinomial Naïve Bayes. Cụ thể trong bài toán của chúng ta như sau:

Gọi C là tập hợp các lớp của văn bản (C có 2 phần tử +1 và -1). Gọi t_i là một văn bản mới đến. Ta chọn xác suất để t_i thuộc vào lớp c_i lớn nhất. Xác suất này được tính bởi công thức:

$$\Pr(c \mid t_i) = \frac{\Pr(c).\Pr(t_i \mid c)}{\Pr(t_i)}$$

Với $c \in C$

Chú ý:

Pr(c) được tính bằng tổng số văn bản của lớp *c* chia cho tổng số văn bản của tất cả các lớp.

Khi tìm giá trị lớn nhất của $Pr(c|t_i)$ ta có thể bỏ qua tính $Pr(t_i)$ do không đổi khi so sánh.

Xác suất $Pr(t_i|c)$ được tính bằng công thức:

$$\Pr(t_i \mid c) = (\sum_n f_{ni})! \prod_n \frac{\Pr(w_n \mid c)^{f_{ni}}}{f_{ni}!}$$

Chú ý:

f_{ni} là tần suất từ thứ *n* trong t_i.

$Pr(w_n|c)$ là xác suất của từ thứ *n* khi cho trước lớp *c*.

Thay $(\sum f_{ni})!, \prod f_{ni}!$ là α, ta có công thức $\Pr(t_i \mid c) = \alpha \prod_n \Pr(w_n \mid c)^{f_{ni}}$

3 KẾT QUẢ VÀ THẢO LUẬN

Để đánh giá hiệu quả của phương pháp đề xuất, chúng tôi đã thực hiện cài đặt giải thuật MNB (Lewis & Gale, 1994) (mô đun phân loại ý kiến trên Twitter), sử dụng ngôn ngữ Python và thư viện wordnet NLTK của nó, đồng thời chúng tôi đã thay đổi cấu trúc chương trình thích hợp với cách lưu trữ tiết kiệm bộ nhớ. Chúng tôi sử dụng mô đun biểu diễn dữ liệu theo mô hình túi từ BoW (McCallum, 1988). Ngoài ra, chúng tôi cũng cần so sánh MNB với một giải thuật SVM chuẩn, được sử dụng phổ biến trong cộng đồng máy học là LibSVM (Chang & Lin, 2011).

Về dữ liệu thực nghiệm, chúng tôi sử dụng tập dữ liệu được sưu tập bởi [Go *et al*., 2009] được lấy từ các API thu thập theo định kì trên Twitter. Các tweets được chép trong khoảng thời gian từ ngày 06/04/2009 đến ngày 25/6/2009 với 72 chủ đề thuộc nhiều lĩnh vực: mua bán, kĩ thuật, âm nhạc, khu vực,… Kết quả ông thu được 1 triệu 6 tweets với 8000 bình luận tích cực và 8000 bình luận tiêu cực không trùng nhau.

Bộ dữ liệu 1 (bộ dữ liệu gốc): 15.000 bình luận được lấy ngẫu nhiên trong bộ dữ liệu 1 triệu 6 của (Go *et al*., 2009).

Bộ dữ liệu 2: là bộ dữ liệu gốc được xử lý biểu tượng cảm xúc.

Bộ dữ liệu 3: là bộ dữ liệu 2 được xử lý từ viết tắt.

Bộ dữ liệu 4: là bộ dữ liệu 3 được xử lý mạng ngữ nghĩa.

Chúng tôi sử dụng nghi thức kiểm tra hold – out để đánh giá hiệu quả của 2 giải thuật phân lớp, lấy ngẫu nhiên 2/3 tập dữ liệu để học (10000 bình luận) và 1/3 tập dữ liệu kiểm tra (5000 bình luận), thực hiện trên cùng một tập dữ liệu mẫu.

Chúng tôi tiến hành so sánh kết quả dựa trên các tiêu chí như:

TP: tổng số phần tử của lớp tích cực được mô hình phân lớp là tích cực.